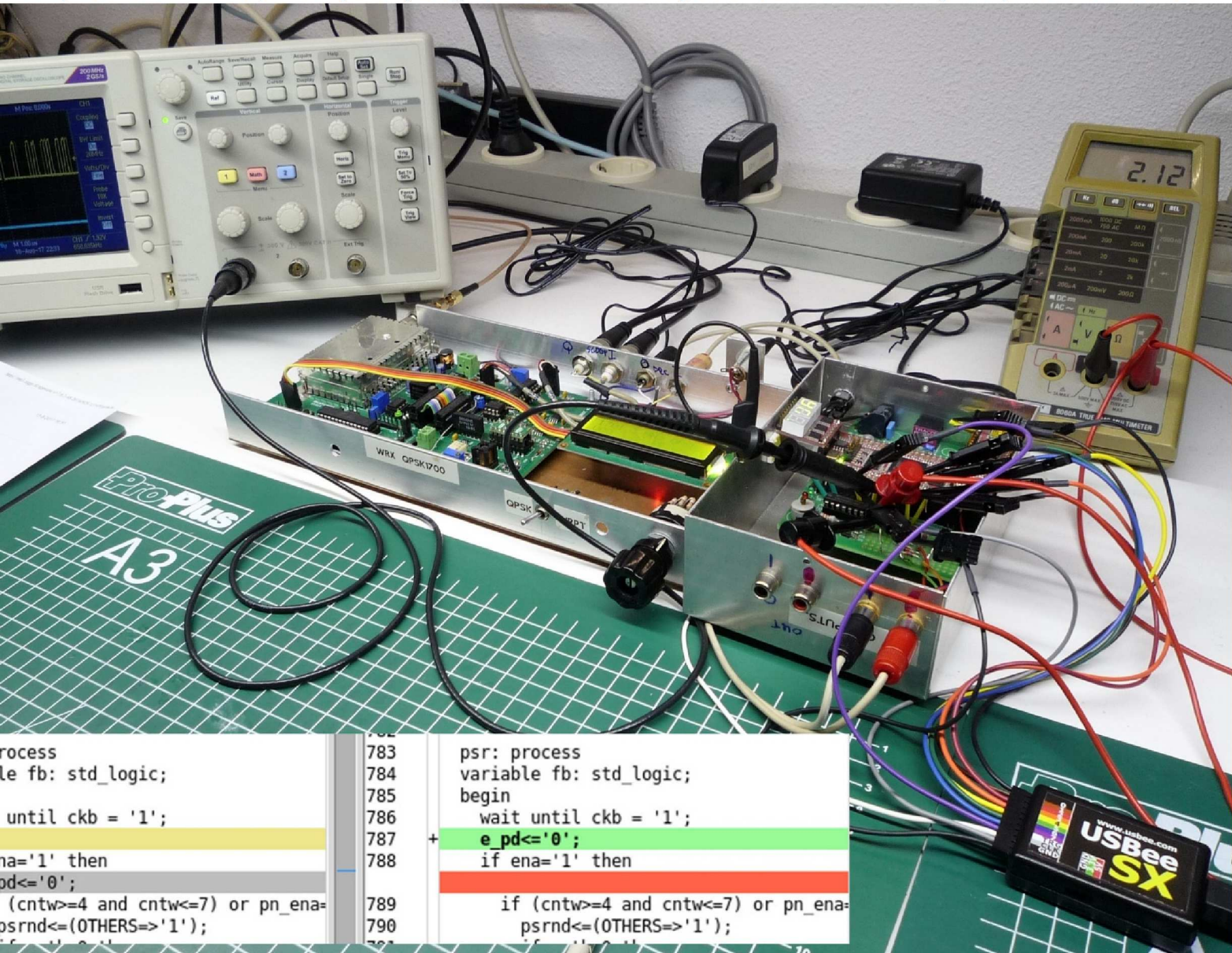




# DE KUNSTMAAN

September 2017, 44e jaargang nr. 3

Uitgave van de Werkgroep Kunstmanen



In dit nummer o.a.

**FT232H**

**Fengyun**

**en nog veel meer**

Dear member,

This pdf contains translated articles of our Dutch magazine “De Kunstmaan”.

Translation for each article is mostly done by the author, using Google Translate. Mostly some corrections are done afterwards. But for sure these translations are not perfect! If something isn't clear please let u know.

Figures are added as much as possible, so the “paper” (Dutch) magazine is not always needed to have at hand.

Internet links mentioned in the articles can also be found at our website; see under menu 'Weblinks' at:

[www.kunstmanen.net](http://www.kunstmanen.net)

I hope these translations will help you to understand the Dutch articles.

Rob Alblas  
werkgroep Kunstmanen  
[kunstmanen@alblas.demon.nl](mailto:kunstmanen@alblas.demon.nl)

## Content

	page KM	page in this pdf
From the Chairman (Preface)	3	4
Weather satellites in Vietnam (25)	5	8
I2C using the FT232H of Adafruit	7	11
Developments decoding Fengyun	11	18
From the library	15	25
AD4351	17	28
Report meeting september 9	21	37
Satellite status	22	39

Photo frontpage: QPSK-receiver + decoder on the “rack”.

At the right side the USBee; see article about decoding Fengyun.

Left-bottom: the offended code where things went wrong, and correction. (See paper magazine for the full photo, partly on the back.)

## Preface

Last meeting was fun and leerza am. This time there was no reading but plenty of time to tune the QPSK receivers.



*The Peter Kuiper receiver (left) is adjusted by Harrie van Deursen (right). Henri Mulder is looking interested*



*With a QPSK signal generator receivers of Job de Haas and Peter Smits compared*

## QPSK receivers

At present five members have the working receiver, six members still have to build the receiver. It is important that we receive feedback about the built receivers. It appears that there will be minor changes to come in the design by variations in components. It appears that the 34063 power regulator Texas Instruments, On Semiconductor still not equal.

Several members have signed up for the QPSK receiver but have not yet received the components set, because the parts are not available. I'm ordering a second round of the organize. Before I do this I want to make sure there are no issues that we still need to adapt to the design. So for those who still have to build one, go ahead and share your experience! With two evenings the receiver must be ready.

## **Power supply**

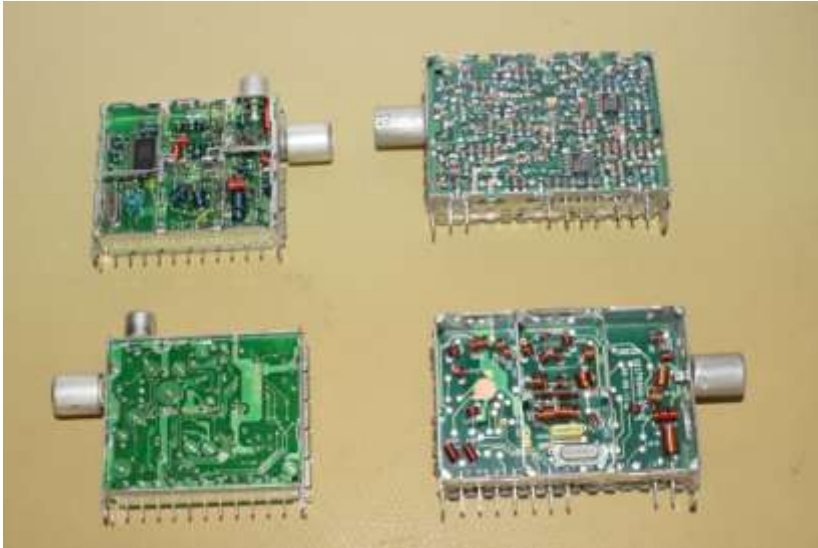
The original QPSK receiver design was based on a stabilized 12V power supply. In the receiver this is brought to a regulator to 5V. We consider allowing the receiver to work on a 5V USB power supply, cheap and universal.

Also for the downconverter and HRPT decoder we can look at USB power. Use a high-quality supply. Cheap power supplies can give a lot of interference and are sometimes life-threatening. To be continued.

## **UV916**

Brings me to the UV916. Here too, there are big differences between them. It seems to be a generic component and certainly not necessarily Philips. There are differences between the antenna terminals: tulip and IEC. There are also differences in the chipsets used: Philips and Motorola. Lots or few inductors. There are also 5V versions. More serious is that it appears that not all UV916 tuners suitable for QPSK reception, but only for HRPT.

So I am again going to look on the internet what's for sale, and that is a lot. So I found the UV1316 tuner, this is (according to the datasheet) from 2004 and a lot more modern. The availability of tuners is a problem child. Hopefully this one does it as well as the UV916 and we have more choice.



*Are these our new tuners? Left the UV1316 and right the UV916 with Philips chips!*

## Filters

The presentation of Jos Disselhorst during the May-gathering of interdigital filters is doing the milling machine at Hendrik running. He has created two new filters. Do the filter from two angle stock, then this has the advantage that you do not have much aluminum milling off like a solid aluminum block. Job has a filter made from an aluminum tube of 50 x 30 mm with a 3 mm wall thickness. The latter is relatively easy to build.



*Interdigital filters Job de Haas. Right top of the model from aluminum tube profile*

## **Jaeger the rotors**

In the previous edition of the Kunstmaan, Harrie wrote about Jaeger rotors. In the Netherlands they do not get, but in England. Via the email I contacted Pulsat.com. It all seemed reliable and ordered two rotors delivered a few days later. They look like NOS (new old stock), probably they have long been in a damp shed. See if I can find some time to get started.

## **Direct LRIT broadcasts**

Eumetsat will stop in April 2018 with the direct LRIT broadcasts. Important considerations are the shortage of bandwidth. The successor / replacement of the old American GOES East (also passed by MSG as LRIT) things get a little tight. And further natural cost considerations.

This direct broadcasts are unencrypted every three hours, and thus without the SKU (Station Key Unit) Eumetsat. Too bad they stop LRIT, I made still hard for me unencrypted broadcasts every three hours. First they were at six o'clock. Within Eumetsat I am known as "3-hour Ben" ... Also a pity that this goes a good test signal. Adjusting the helical went perfectly on the LRIT signal. LRIT data is still on the FTP site Eumetsat available.

In this Kunstmaan again many worthwhile articles. Rob writes about the new developments in the decoding of the Metop / Fengyun. Thus, it is now added Reed-Solomon error correction, the number of interference lines is thus decreased.

Fred writes about his experiences in Vietnam, where as here a lot of rain falls. Our librarian has shaken off the paper dust and shares his views with us. He brings us the much needed relativation.

Last summer I got a new USB interface played the FT232H. This comes in the HRPT decoder. I also have the ADF4351, an integrated PLL VCO Analog Devices to the test and it worked. In this Kunstmaan about this much more.

The next meeting is on November 11 and on November 4 we have a stand on the Day for the Radio Amateur. Come all!

### Summary

My experiences with weather satellites etc. in Vietnam.

### General

Goh, the 25th episode of my quarterly gossip. Where has the time gone. Time flies ...

...though the fun this time was a bit less. Unfortunately, this is a brief contribution, due to circumstances. I have been less mobile by back problems, something that is a more or less annual ritual. Only this time it took longer than usual. Yes, it's going to be reasonable again, thanks.

In addition, rain and thunder were the breakers. When it's raining, I do not rotate the rotors to prevent from possible water in the rotors. And thunder is already clear: the roof is quite high relative to other objects, so I do not take any risks. All antennas etc. go loose. And yes, it took place mostly in the afternoon, as did the NOAAs.

And of course, as always, we had to spend a lot of time in repairs, related to the Vietnamese "quality" of many things. And I suspect that Murphy is also spoiling things so here and there.

OK, back to business.

### HRPT

#### Satellite PC

I have been busy with my satellite pc. It's already rather old and starts to become decayed. There will be something new in the not-so-long term. But for the time being, I can still keep it alive with some tricks.

I started with Weather Display, a program that displays the results of my Davis weather station and has been giving problems for some time now. After an afternoon trial-and error it suddenly worked again. I immediately secured the cfg file to another system.

The APT part, which works fully automated, was checked and I did some file management. It can do its job again for some time.

Then I set up the PC for HRPT. I installed Wsat, xtrack and HrptReader and did a setup based on the laptop, on which I'm still testing. In addition, I installed AutoHotKey on the PC with some related items, checked the scripts and made some minor adjustments. Works now also fully automated.

Somewhere in the next weeks I will pull all cables down the pipe to my shack and immediately check the other cables if they are still fine after the attack of the rats.

The only thing I do not know yet is whether the USB connections for the APT and HRPT receivers might bite each other or if the PC is too slow to process two signals at a time. In that case, I will expand my AutoHotKey script with a few lines, which will turn off the APT reception at the start of HRPT and turns on automatically afterwards. Cannot be a really hard job. I hope.

The aim is still an as much as possible automatic procedure with occasional manual input (H):

- (H) Power HRPT installation on (1 switch)

- (H) Run AHK start script
  - Correct time
  - Start Wsat and xtrack
  - Pick up Keplers
  - (H) Set the receiver to the correct frequency
  - Receive satellite
  - Start HRPTReader and create a number of different images
- (H) Run AHK stop script
  - Quit all executed programs at once
- (H) Power HRPT installation off

### Rotor

Just before submitting this article, I received via Lazada from China a portable-waterproof-emergency-space-rescue-thermal-mylar blanket, as the thing is named so beautiful on the site. This must come around the rotors, hoping that the reflective material will keep the heat somewhat away. Time was too short to test, so probably next time.

## **Miscellaneous**

### Bending machine

Many members know the bending machine, which Peter Smits has built. The Astro-forum shows a heavier type, complete with origin and issues, this version gives. [1]

You may subscribe to become a member of the Astro-forum but that's free.

### Super-bino

In another topic, someone describes the construction of an aluminum super-bino. For those who want to get many ideas about how to build in aluminum, an absolute must. [2]

### NASA computer

In a garage in Pittsburg a computer with data tapes from the Apollo era has been discovered. The owner of the garage died in 2015 and the one who cleaned the garage came across the computer with 300 tapes from NASA's Goddard Space Flight Center from 1969 to 1972. [3] This post contains some links to interesting websites and other information. For the older youngsters among us great stuff.

### QR codes

Fortunately, there are more people who see the utter uselessness of those paint clovers. And no, an option as on the back of this extraordinary magazine is not included in the 5 useful applications [4].

### ISS

Take a look at the ISS with Google Streetview [5]:

### Cassini

Cassini is busy with his latest rounds of Saturn and puts the made pictures online every time. There are several amateurs who have edited images. Whoever wants to try it, see [6].

### Solar eclipse

A beautiful photo made by the Lunar Reconnaissance Orbiter can be found on [7].

## **References**

- [1] Bending machine, see website
- [2] Superbino, see website
- [3] NASA computer, see website

- [4] QR codes, see website
- [5] ISS, see website
- [6] Cassini, see website
- [7] Eclipse, see website
- [8] Rescue blanket, see website

## **I2C with the FT232H of Adafruit**

### **Summary**

This article describes using the FT232H-module from Adafruit for I2C-communication with electronic devices. With a python-script running on your pc you can control directly I2C-devices.

In the future the FT232H module from Adafruit will replace the FT245RL module from Sparkfun (which is now obsolete) used on the HRPT-decoder.

### **Introduction**

In the HRPT-decoder the module of FT245 Sparkfun is used and is not available anymore. We are looking for an alternative. Dirk de Schutter alerted me to a module of the Elektor FT232H FTDI. Unfortunately, this module was no longer available and self-construction was very expensive (the pcb costs 25 euros). Further, search learned that Adafruit also has a module to the FT232H and that it is easily obtained in the Netherlands, see the links [1].

Only a small change in the bit - file for HRPT decoder needed to work with this module. This adjustment was for the SparkFun module also required, but that worked well. The "only" thing needed, is another print for HRPT decoder because FT232H of Adafruit not pincompatible is that of Sparkfun. The new pcb is not the subject of this article, this comes later as the new decoder is fully crystallized.

In addition to replacing the FT245, the FT232H can also be used to control I2C components directly from the computer.

### **Description of the FT232H**

Here I'm going not going to give a full description of the FT232H. The data sheet and the website of FTDI can do this much better.

Some details: the FT232H has a 480Mbps USB interface. The interface is serial or parallel, the latter we use in the HRPT decoder.

You can also create JTAG, SPI, I2C and UART interfaces. This occurs in the Multi-Protocol Synchronous Serial Engine (MPSSE), see figure 1.

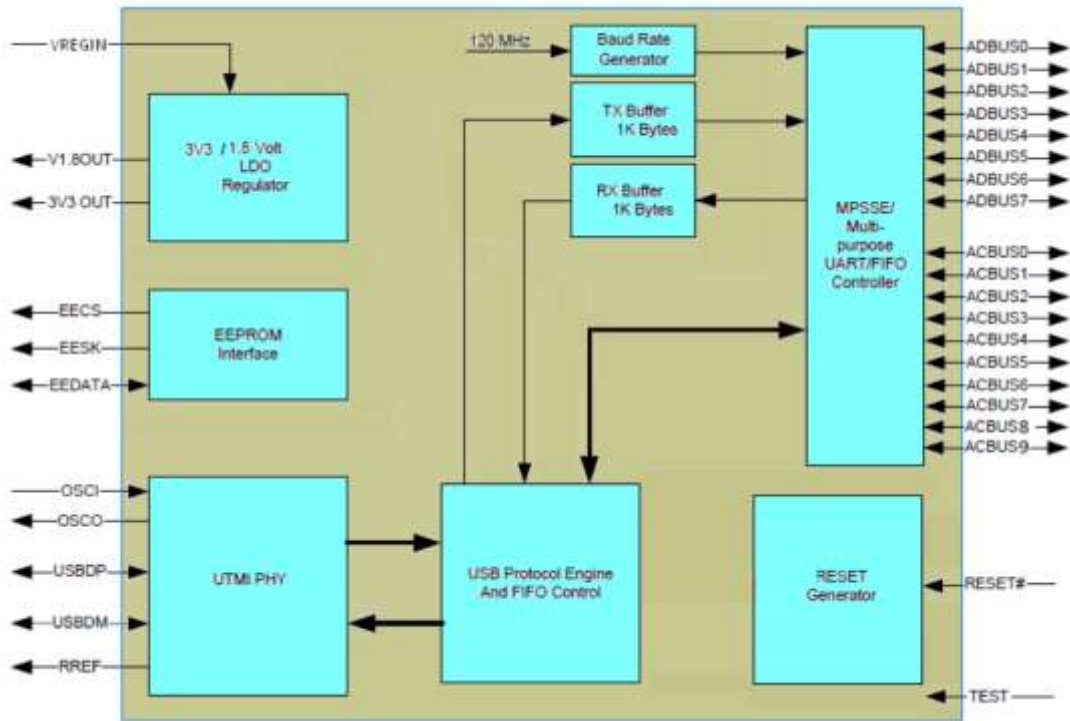


Fig 1. On the right hand with which the MPSSE including I2C and SPI.

The chip is in a 48-pin LQFP or QFN casing supplied, difficult to solder. Therefore, it is good that Adafruit has made a module, see also their website [2].



Figure 2 FT232H module of Adafruit. The chip has an on-board regulator 3V3. Unfortunately, this is not available on the pins.

## **I2C via the FT232H**

On the website [2] of Adafruit is a detailed description of this module, with many examples how to "talk" with your computer via this module via I2C and SPI.

The I2C protocol years ago invented by Philips to communicate with the various components in TVs as simple as possible. This is based on a clock - and a data signal. The FT232H is the master initiating communication with the I2C part, the slave.

Want to "talk" to an I2C component you can do this via do a microcontroller. In WRX1700 and the QPSK receiver we do this. The Arduino talking to the UV916 and the LCD display via I2C. The Arduino's has even a special library designed.

It's harder to talk to an I2C component from the computer. You should first be able to send information to the Arduino, which then sends it to I2C. This solution requires a lot of programming on both the PC and the Arduino side.

The FT232H module and Python scripts would be a lot easier communicate with I2C components directly from the PC. Python is called a scripting language that allows you to program / test something quickly. Python is designed by Guido van Rossum (a Dutchman) at the end of the eighties. Python runs on Linux, Windows, Mac, AS / 400. Now I would not stabbing the praises of Python but it still seems to be the scripting language to be right now. It is also the language used for the Raspberry Pi. Maybe that's a topic for another article.

## **Software**

Still, I've been working to install everything and talk to it. In this article, I describe the installation under Windows.

Several software packages to be installed:

- Driver for FT232H
- Python 2.7. Although I have a 64-bit computer, you must install the 32-bit version because otherwise communicating with lib - usb not working. According to the website, newer versions of Python seem to not work.
- Sad driver
- libftdi
- python library for Adafruit FT232H

See in front of a extensive Description of software installation website of Adafruit [3] true step by step with screenshots is becoming explained how to use all software install.

## Connect I2C devices to the FT232H

If you have installed all software, the hardware must be connected to the computer via USB.

- The pins D1 and D2 on the module must FT232H to be connected
- On the interconnected pins D1 and D2 is, the SDA signal (data), and these must a pull-up resistor of 4k7 to 5V
- Pin D0 on the FT232H module is SCK signal (Clock), and this must receive a pull-up resistor of 4k7 to 5V
- To the I2C part, SDA, SCK, + 5V and GND.

## Python I2C scripts

Below I will discuss a few Python scripts for communication via I2C. A very nice script to begin with, that all scans the I2C-devices on the bus, the addresses found are displayed.

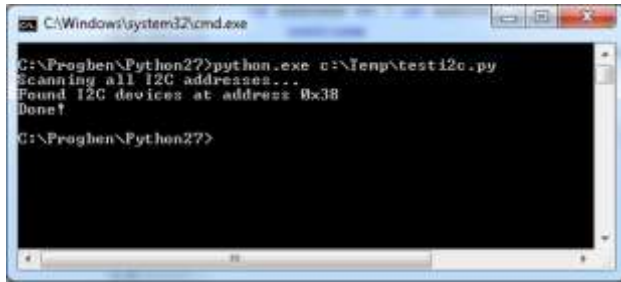
Software is correctly installed above or below your script in a text editor write and save the extension. py.

```
Import Adafruit_GPIO.FT232H as FT232H
FT232H.use_FT232H ()

ft232h = FT232H.FT232H ()

print 'Scan all I2C addresses ...'

for address in range (127):
    if address <= 7 or address >= 120:
        continue
    i2c = FT232H.I2CDevice (ft232h, address)
    if i2c.ping ():
        print 'Found I2C devices at address 0x {0:
02X}'.format (address)
print "Done!"
```



*Figure 3 The script log t back to 0x38 that it has found an I2C device*

Then I wanted to see if it is possible to control four LED displays connected to the SAA1064 (I2C display driver) to drive. This turned out not so easy because the Adafruit library assumes that writing to an I2C device in "registry value" goes combinations. The SAA1064 (and many other chips) now does not addressable registers. After a moment's sitting browse through the library FT232H turns it possible to fill the registers without addressing them.

Below is a script to control the SAA1064.

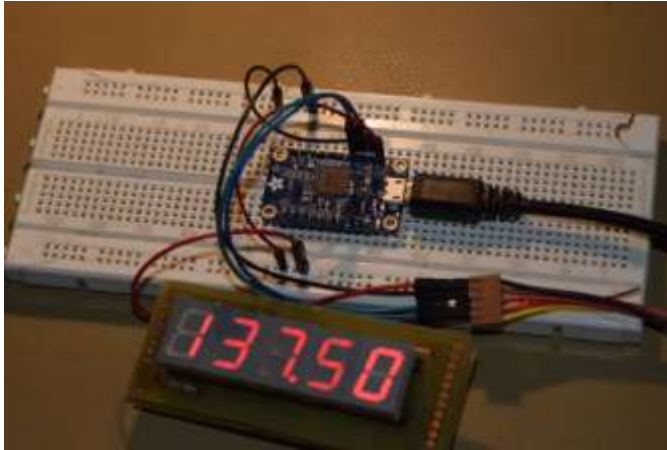
```
Import Adafruit_GPIO.FT232H as FT232H
FT232H.use_FT232H ()

ft232h = FT232H.FT232H ()

i2c = FT232H.I2CDevice (ft232h, 0x3B)

i2c._transaction_start ()
i2c._i2c_start ()
i2c._i2c_write_bytes ([i2c._address_byte (False), 0x00, 0x37, 0xF2,
0xE1, 0xB6, 0xFC])
i2c._i2c_stop ()
response = i2c._transaction_end ()
i2c._verifydata_acks (response)
```

The lower part (starts with the line `i2c._transaction_start`) controls the I2C communication and consists of six lines. In line with the instruction "`i2c._i2c_write_bytes`" we write multiple bytes to the SAA1064. The first two bytes are the instructions - and control byte, the last four to fill the display. More information about the SAA1064 is in the datasheet.



*Fig 4 A 4-digit LED display controlled by a SAA1064. The first digit (1) brand t always.*

## ADF4351

Elsewhere in this Kunstmaan, my article is about driving the ADF4351. This is also possible via the FT232H, which also supports SPI.

```
import time

Import Adafruit_GPIO.FT232H as FT232H
FT232H.use_FT232H ()

ft232h = FT232H.FT232H ()
spi = FT232H.SPI (ft232h, cs = 8, max_speed_hz = 30000, mode =
0, = bitorder FT232H.MSBFIRST)

time.sleep (1)

spi.write ([0x00, 0x58, 0x00, 0x05]) #Write register R5
spi.write ([0x00, 0xCC, 0x80, 0x3C]) #Write register R4
spi.write ([0x00, 0x00, 0x04, 0xB3]) #Write register R3
spi.write ([0x00, 0x00, 0x4E, 0x42]) #Write register R2
spi.write ([0x08, 0x00, 0x80, 0x11]) #Write register R1
spi.write ([0x00, 0x50, 0x00, 0x00]) #Write register R0
```

The order in which the bits are written must be MSB. In the last six lines, the data is written to the registries. This is a literal translation of Figure 6 in the ADF4351 article.

In the library of Adafruit refer the pin numbers 0 to 15 to D0 to D7 and then from C0 to C7.

The connection is as follows:

- On D0 pin is the clock-signal
- On pin D1 is the data signal
- Pin C0 is the slave-select signal. In this example, code is "cs = 8"
- The GND must be connected

## UV916 control

Of course, the HRPT-QPSK receiver UV916 can also be driven in this way. However, we can not replace the Arduino in order to control the entire receiver. The Arduino is used for reading the VCO and RF level and controlling the LCD display.

## Concluding remarks

The FT232H is a versatile chip that can be used for more business than just serial or parallel data transport. Besides I2C and SPI as discussed in this article you can also use the IO-lines eg. driving an LED sending or reading input.

The HRPT decoder we will encounter him, but that later more.

## Links

[1] Web shops that sell FT232H of Adafruit

<https://www.kiwi-electronics.nl/>

<https://www.sossolutions.nl/>

<http://us.mouser.com>

[2] Description / manual Adafruit

<https://learn.adafruit.com/adafruit-ft232h-breakout/overview>

[3] Software Installation

<https://learn.adafruit.com/adafruit-ft232h-breakout/windows-setup#>

## Developments decoding Fengyun

In the previous KM (No. 2, p. 15) I did describe the Fengyun decoder. The decoder seemed to work well in the first place, but later on a few problems popped up. The first problem has to do with the second sync word. As explained in KM2, the beginning and end of a line at METOP is clearly defined. See fig.1, identical to Figure 2 in KM1 of this year, page 14.

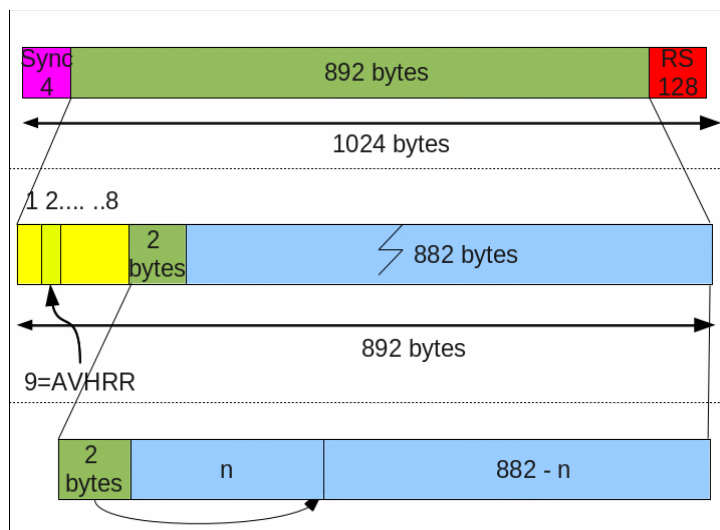


Fig. 1. Structure frame METOP and Fengyun.

Shown from top to bottom:

- the full frame
- header part, position 2 contains a code word for the type of data; '9' means: AVHRR.
- the data part.

In this last section, the number value of the two first bytes indicates where a new line begins (and the previous line ends); code "2047" indicates that the entire data portion of 882 bytes is part of the same line. Subsequent frames thus first contain a block where the 2 bytes indicate the beginning of a line (a number between 0 and 881), then a number of frames with the numerical value "2047", then a frame in which the end of the line is indicated. Concatenation of the marked bytes results in the complete data block of a single line, in which the 5 METOP channels are located. Thus, the beginning of a line can easily be recognized by the 2 bytes at the beginning.

With Fengyun, the structure is approximately equal; Byte 2 in the middle bar of Figure 1 also shows the type of data. That is, for example, code 5 for "VIRR day".

However, the first two bytes in the bottom bar are always 2047. This means that no beginning and end is indicated on the data stream; as a result, start of a line cannot be derived from these bytes.

Instead, an additional 3-bytes sync word is added at the beginning of a line:

84 5B F5 (hexadecimal)

This codeword must therefore be detected by the software. Initially, I assumed that this syncword is added on byte-base so that the above-mentioned data word is easily recognized. Data received later on showed, however, that the sync word may also be bit-shifted. A more complicated detection is therefore required.

Fig. 2 shows how the channel data is added. A total of 26047 bytes each line is available. With 2048 10-bit words and 10 channels,  $2048 \times 10 \times 10/8 = 25600$  bytes are required. The remaining 447 bytes are for calibration data ( $26047 - 25600 = 447$ ). These data bytes are placed directly after the 3-byte sync word and may therefore also be shifted a random number of bits, together with the sync word.. Back-shifting is possible using the  $3 \times 8 = 24$ -bit sync word. Hereafter, like with METOP, the 10-bit pixel words can be derived from the bytes. Except that here are now 10 channels instead of 5.

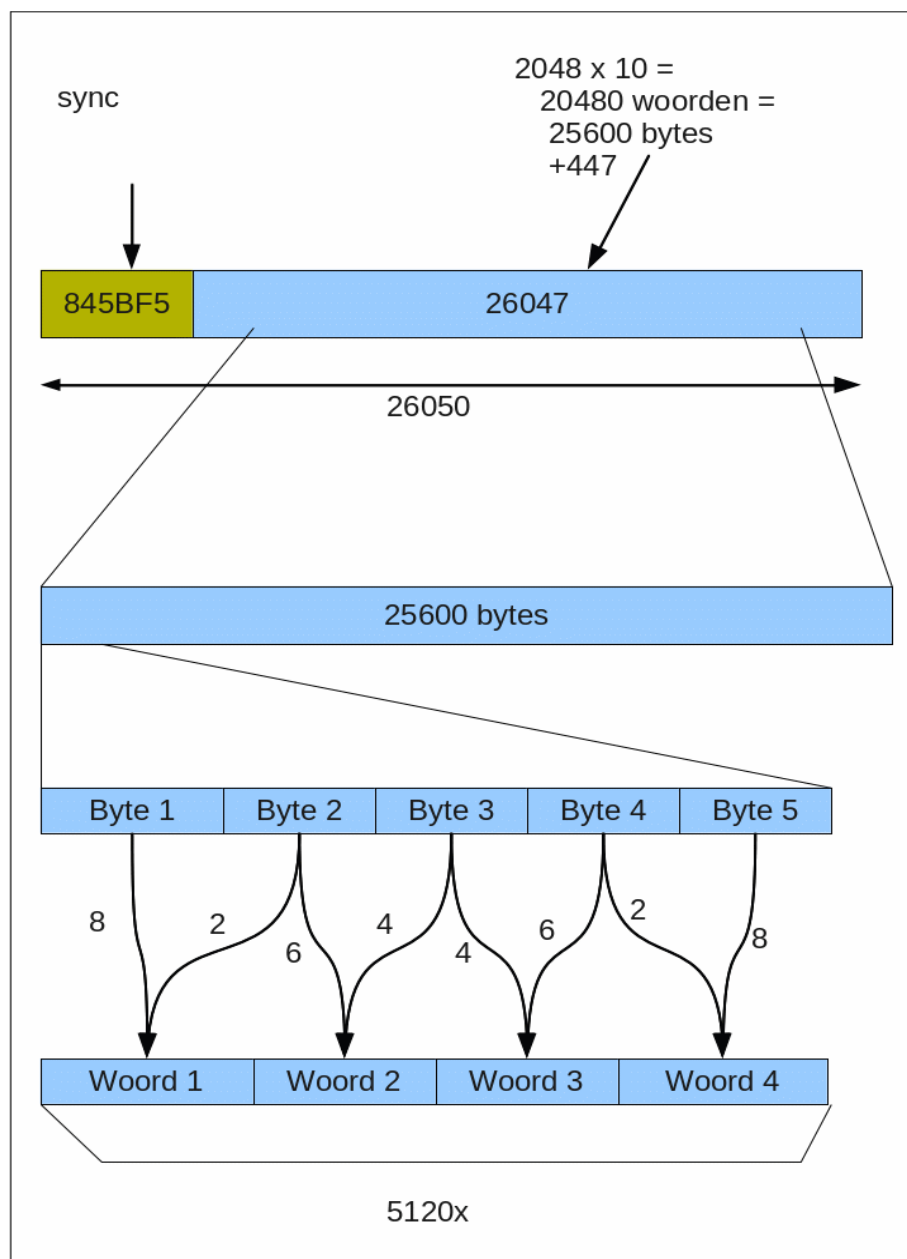


Fig. 2. Extract 10-bit words from Fengyun bytes.

This solves the problem with the shifted sync word.

Nevertheless, there were frequent problems with recording Fengyun; sometimes it was OK, other times the recording was unusable. In order to investigate where the problem occurred, I put out some internal signals in the decoder on the pins. That's easy to do with an FPGA; you can make what you want, and thus also output signals that are normally not accessible. The FPGA has enough pins. Using the logic analyzer USBee, the course of these signals has been recorded for a short time. This showed an error in the decoder design. After correcting, the decoder works well. That USBee has proven its usefulness. For a description of this handy device, see "De Kunstmaan" 2014, April, No. 1 (directly available as pdf via our website).

## Error correction

As previously indicated, the received data contains additional bytes for error correction. This so-called Reed-Solomon correction takes place on fixed block lengths. Figure 3 shows that there are 4 blocks of 223 bytes per frame for 1024 bytes each. 32 bytes are added to each block; those blocks are pushed to the end of the frame. Correction is then done on block b1 with the additional bytes f1, block b2 with f2 etc. The result is that errors in b1 to b4 can be corrected to some extent. After that, the bytes in f1 to f4 are no longer necessary.

In the first experiments, I dropped this part. Then, f1 to f4 are immediately discarded and you will work with the uncorrected b1 to b4. That was enough to test the decoder and software. Later I added the correction to the wsat program, which gives a significant improvement. Because the correction takes place immediately during recording, this operation does not take extra time.

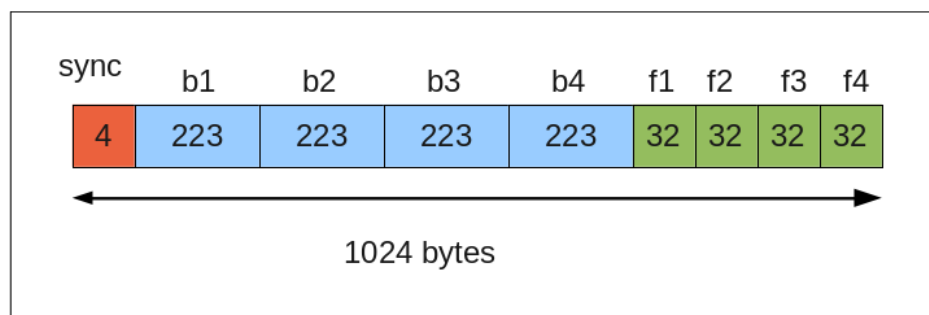


Figure 3. Split frame into 4 blocks with correction fields.

This error correction is also done during "Live" display of the received data.

Note that the correction is done for all received bytes except the 4-bytes syncword (not to be confused with the previously mentioned 3-bytes 2nd sync word!). Correction on the 4-byte sync word is nonsense; if the reception is so bad that the sync word is not recognized then the blocks in Figure 3 are no longer recognized as such, and if the sync word is recognized then that is good enough. (The decoder allows a bit error in the sync word.)

## Different bitrates

With the same hardware, HRPT, Meteor, Metop and Fengyun can now be decoded. One problem is that the frequencies of HRPT, METOP and Fengyun are not simply divisible from a single VCO frequency. That means that especially with switching between Fengyun and another satellite the VCO must be trimmed. Harrie v. Deursen, who has tested the

decoder extensively, has already worn many trimmer ...

The frequencies to derive from the VCO frequency of approximately 21 MHz are:

type	bitrate	Needed division	Used division	difference without trimming
HRPT	665,4 kb/s	32	32	0
Meteor	665,4 kb/s	32	32	0
METOP	2,33 Ms/s (QPSK)	9,138	9	+33 kHz
FY3A, FY3B	2,80 Ms/s (QPSK)	7,604	8	-139 kHz
FY3C	2,60 Ms/s (QPSK)	8,190	8	+61 kHz

Note that Fengyun3-A and B have a different bitrate compared to Fengyun3-C. It can be seen that, based on VCO adjustments on HRPT, the required frequencies for, in particular, Fengyun-3A and 3B differ quite from a whole particle.

One possibility would be to take another trimmer for each satellite, with a selector switch. Not nice.

Another method is to set off the VCO varicap offset voltage, depending on the chosen satellite.

A third solution, which has now been chosen, is the use of non-integer dividers. This is possible by using a combination of a divider and multiplier. Multiplying can be quite simple with a PLL; Figure 4 gives an example of how such a thing could be built.

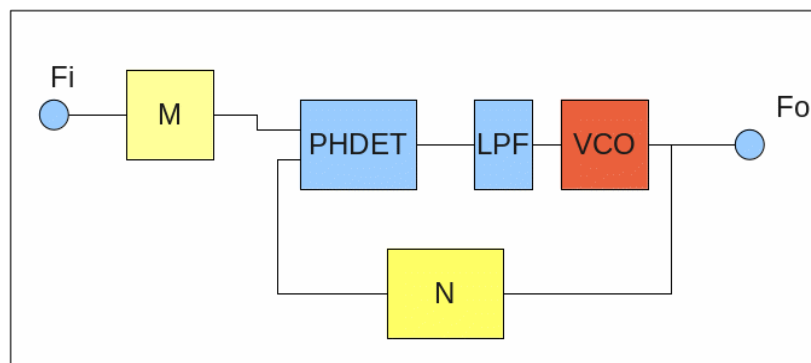


Fig. 4. Dividing using non-integer.

This is a PLL, with a divider N in the feedback loop. The phase detector will adjust the VCO such that the two input frequencies on that phase detector are equal. Because of the divider N, the VCO will then be set to an N x higher frequency.

With a second divider in series with the input we get now:

$$F_o = F_i * N / M$$

Eg. How to make from  $F_i = 3$  MHz a signal with  $F_o = 2$  MHz, a factor of 1.5? Simple, using Figure 4, take  $N = 2$  and  $M = 3$ . Then  $F_o = F_i * 2/3$ .

In this way, using one input oscillator connected to  $F_i$ , all needed frequencies, can be synthesized, that is, approached close enough.

The question now is how this can be realised with the used components. As is known, the decoder is realized in a so-called FPGA, Field Programmable Gate Array. This enables any digital circuit to be realized.

The dividers and phase detector present in Figure 4 can easily be made, however, the VCO is another story because it is an analogue component. Because a PLL is a functionality that is often needed, the FPGA we use includes 4 so-called DCMs: Digital Clock Manager. This allows exactly the functionality of Figure 4 to be realized, although internally, it is completely different. The advantage is that no additional components are needed externally. In the DCMs, the partitions  $M$  and  $N$  must be between 1 and 32.

In order to make it clear now, the initial situation is shown in Figure 5a. Note that the VCO drawn here has nothing to do with that in Figure 4. This is an external one, which is used to lock on the incoming satellite signal.

This VCO at about 21 MHz controls two dividers, one of which is reset by edges of the incoming bitstream  $D_i$ . Different satellites require different sub-factors; see also the table. In addition to the VCO, the LPF (low pass filter) is external, so not in the FPGA. Starting point for the VCO frequency is HRPT:  $0.6654 * 32 = 21.29$  MHz.

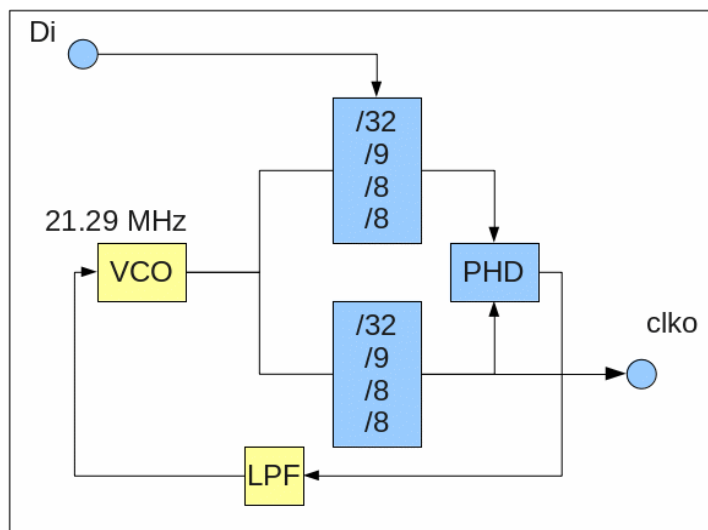


Fig. 5a. The original PLL.

Figure 5b shows the new situation. Important to note that the DCMs are fixed, so there are really two DCMs needed. (The FPGA used contains a total of 4 DCMs.) At the delers, that is something else, which is set to the correct value using the selected satellite. Hence two DCMs which one is chosen there.

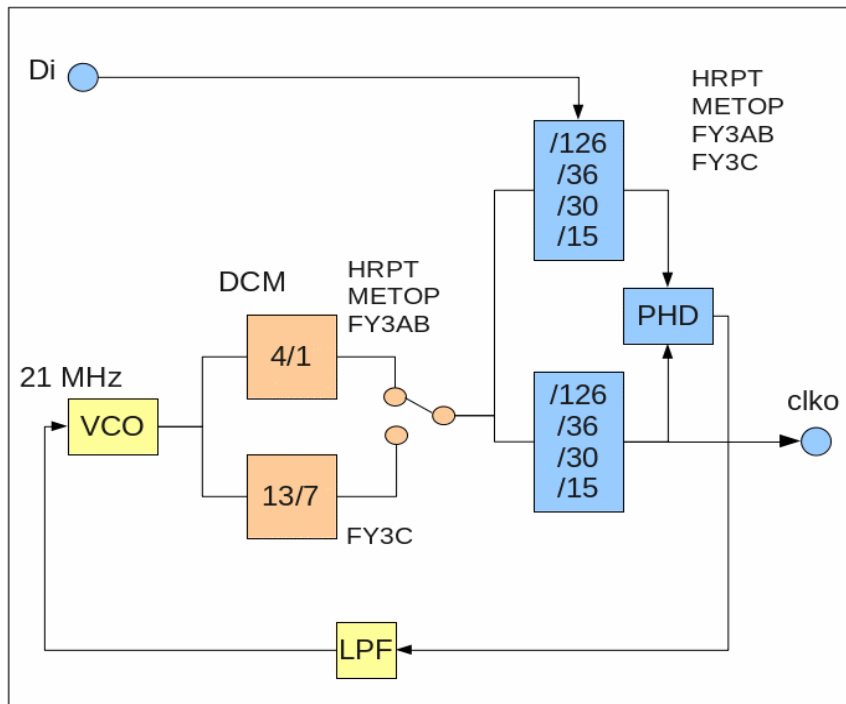


Fig. 5b. The new PLL, with DCMs. Multiplication of VCO frequency, ratio in DCM blocks, and sub-factor in right-hand divisors gives the frequency on clko.

The starting point is now METOP:  $2,333 \times 36/4 = 21$  MHz. The following table shows the new situation.

The maximum frequency deviation is only 1.26 kHz, instead of 139 kHz. This small deviation can easily be handled by the VCO.

type	bitrate	Needed divider	Used divider	Difference without trimming
HRPT	665,4 kb/s	31,56	31,50	+1,26 kHz
Meteor	665,4 kb/s	31,56	31,50	+1,26 kHz
METOP	2,33 Ms/s (QPSK)	9,00	9,00	0 kHz
FY3A, FY3B	2,80 Ms/s (QPSK)	7,5	7,5	0 kHz
FY3C	2,60 Ms/s (QPSK)	8,077	8,077	0 kHz

As a result all mentionde satellites can be decoded without the need of manual VCO trimming.

One point of concern when using the DCMs is to maintain the maximum jitter on the DCM (21 MHz) input signal below 300 ps for proper operation. However, the VCO is driven from a "noisy" environment (the received signal). This does not seem to give any problems for

the time being.

Some additional comments:

- The division factor of the "blue" dividers have been multiplied by (about) a factor of 4. One of the reasons is that the HRPT component must be even. This has to do with the fact that HRPT is split-phase. In addition, this is beneficial for sampling the input signal. This is a story in itself, something for a next time.
- For the new circuit nothing has to be changed to hardware, only the content of the FPGA must (of course) be modified.

## ***Literature***

[1] KM No. 2 2017, p. 15. Decoding Fengyun

[2] KM No. 1 2017, page 13: Direct receipt of Metop part 2

[3] KM No. 1 2014, p. 37: USBee Logic Analyzer & Signal Generator

## From the Libary - Kunstmaan 2017 Vol 3

Dear people,

This summer Mrs. Librarian-in-Chief went on garden trip to England with her friends and other relatives. I think it was fantastic trip! Four full days the house was a garden of pleasure full of striplines, manuals, yagi antenna's, controller boards and cables throughout all of the house. Of course I also took ample time to share my joy about modern technology with you.

The most miserable programming language on earth by far is the language C. This attempt to keep muddling with loose bits in the computer and yet minic a safe remote third generation language (fundamentally incompatible of course) has yielded a patchwork full of brackets, dirty broken loops, exceptions, apostrophes, semicolons, soup slices and ampersands. And stars. A lot of stars. That is why you see stars before your eyes.

No one has ever been punished for this misdeed. On the contrary, the creators have been applauded as visionairs for decades. Language C has also become a springboard for a stud farm of "improvements" as there are: Perl, PHP, Python, Ruby, C-sharp, Java and also C plus plus. I would never have made my keyboard dirty with it, but it happens to be the language with which to control an Arduino and hence our two motors. Now we are left facing bad music.

Sooner or later we have to learn it. At first it does not seem too difficult: a statement like this: `LCD.write ("Good Day Dear Group Members!");` is easy to understand. The trouble is in other things. I tried Structured Programming with C ++ from Bäckman, a free e-book [1]. Good price-performance ratio, but it's also useful? I take the advertising, from companies aimed at young people, for granted and found a book that takes the reader takes along in nice steps. It uses Visual C ++, free as well. Nice book if you want to calculate pi yourself or build your own sorting routines, or more in our field, decoding routines for the reception data. However, if you only go for the Arduino, I think studying sketches will suffice.

In the April issue of Electron we find a weird story about the use of a high-voltage mast as a short-wave antenna. Capacitive coupling is not allowed (too dangerous) and so it must be done inductively. Ferrite ring around the foot of the mast: fill a garden hose with iron filings and perform inductively coupling. I stopped reading; I can no longer stand that eternal muddling around on those direct current wave lenghts. I am happy that normal people start living from 137 MHz onwards. That said, I read a comic emergency call for volunteers, for keeping of meetings, lectures and teaching. I almost wanted to sign up; I am also here for you already active. But yet I have sent the VERON an email that I noticed it just in time and see it as a nice april fools joke.

Found in Scientias internet magazine: a photo of a lightning blast, just before the nose of the photographer [2]. It looks like a luminous fat tree trunk. The last few weeks we have here in the region had a few heavy lightning storms and here is one going on right now. That brought me to the next thought: Do we have any defense as is something stuck in a receiving dish? Chances are not big, because we are not high on buildings, but nevertheless. You can forget about the feed antenna, but the other equipment? Can we switch behind the over to fiber and so protect the equipment further downstream?

USB scope. I was to try my new LabNation USB scope [3] on an old XP. Not done yet. I can do not worry, because it is true that the prices of those scopes fall sharply and performance rises. Soon it is not even worth the effort. But yes, that's how technical developments go. Soon, oscilloscopes are for sale at the price fighter in household items, somewhere between the T-shirts and the barbeque stuff. The idea. The nightmare.

Technical idea as seen at a colleague, hobbyist and astronomer. He has a telescope with stepping motors in polar alignment. To suppress vibrations and reduce engine noise, he sends the square wave through a low pass filter and then you get rid of most trouble. It takes more energy, but avoids mechanical wear and noise. Maybe an idea for here (if not done already of course)? I still have the question to be answered to filter before or behind the controlling FETs.

MOSFETs, yes. They can not be trusted. When I first saw them I was impressed. Infinitely high  $R_i$  and low  $R_d$ s that no BJT get close to. Until I learned that you should not be naive; the gate current can get up to 1 ampere due to the input capacity, and that's pretty much for a cap of 300 pF. One night I was poking around on the web and read about more possible misery for your motor control. Unwanted oscillations, Miller capacity, interference for driving controller, slow recovery of the parasitic diode, both bridge FETs conducting simultaneously; just choose your own favorite trouble from this (and more) in Why MOSFETs fail [4].

Photos and movies (a little bit flavoured with extra color and relief of course) of Pluto and Charon, offered to us by NASA [5]. At last some more material of the fly-by of already 2 years ago. The probe is already on its way to a next rock, the MU69, with 50,000 kilometers per hour. You and I get fined for less, but fortunately you will not find it there speed checks. less than 1000 million kilometers to go to New Year's Day 2019. That will be a success. I'll let you know.

ESA (European Space Agency) public visitors open day. It will be again this year held on October 8th, free, but apply soon [6] otherwise you will not be allowed. Already I promised to report to you earlier about this event. I found out that the visit requires a lot of fitness which I do not have, stumbling between human masses, hearing trouble, look at into lamps, take notes; it did not work out too well. You are amazed at what's

going on, varying from mothers with baby cars to youngsters with a heavy packing on the back. What I do remember: A very typical antenna. It seemed like it most on a printed circuit board with headers in the shape of drops with different directions. It was intended for space probes. Someone who can explain this? Otherwise, I will inquire in October. If I succeed in getting to the presenting speaker.

In August America had a solar eclipse. That's not very special, but some people there had to cry. I do not understand; the bite out of the sun is really coming back. I particularly liked the people in America though who have calculated and observed and filmed a simultaneous transition of ISS in front of the sun. We also do visual observations at our club, I like to go out in the garden and observe ISS's passes. This is a much more advanced. Just look at the searching picture. If you want to see ISS in half a second's passing, go to [7] at 3 minutes 31.



ISS shoots before the sun during the solar eclipse

It's swinging in our club! All kinds of things do happen. New young members just take a home built antenna, put them in Utrecht in the garden or hang it in a tree (the first time that I ever saw a tree doing something useful) and connect a laptop. Flawlessly the sounds from the NOAA come to us. Never before, our chairman told me, at a meeting a satellite got received in real life. Applaus! I'd rather see this than those types of people with a thick layer of paper dust on their heads, knocking sour comments into their keyboard. If there's just one person that dislikes that kind of people, it is ... uhh ...

your librarian

References

See our website

## **ADF4351, a synthesizer with integrated VCO**

### **Summary**

This article describes our experiments with the ADF4351, an integrated synthesizer and VCO ranging from 35 to 4400 MHz. The software from Analog Devices is used together with a Cypress microcontroller and the ADF4351 module.

### **Introduction**

In my experiments with band converter I was on the site of Mini-Circuits looking for VCOs. I found out that they surely stop some models because some parts that they use are EOL. What they have in stock and will sell them depending on whether they will develop new VCOs again, see [1]. It is a long list (275 pieces) of VCOs disappearing, so time to invent something else.

I had already noted the ADF4351 from Analog Devices. This is an electronic component where the PLL including VCO. I wrote about the ADF4106 PLL coupled with the VCO ZX95-2420 S + Mini Circuits in Kunstmaan of September 2015. The ADF4351's all this in a single chip, very comfortable!

I had even a PCB designed for but had never finished. The ADF4351 is another part to be soldered to the bottom .... Luckily I found on eBay a module that can bring you right to work. In this article I describe how the development board from Analog Devices can be recreated with parts that are for sale on eBay.

### **ADF4351**

For all possibilities, please see the datasheet [2]. The main features are:

- An output frequency of 35 to 4400 MHz
- An integrated PLL, this is a combined fractional-N and an integer-N synthesizer
- A programmeerbare output level
- I and three-wire interface

It is very convenient that Analog Devices has released a tool that you simply can set all the capabilities of the chip [3].

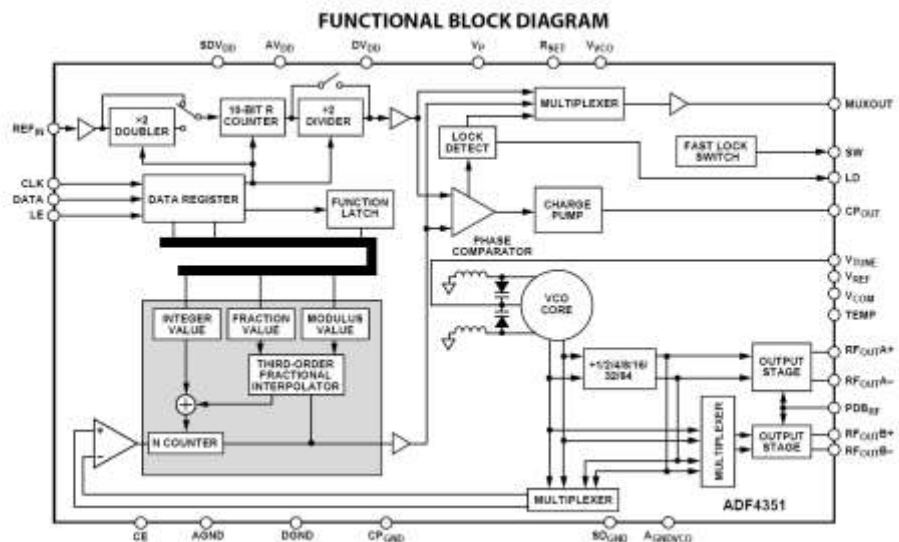


Figure 1 Block diagram of the ADF4351. Below the VCO with the two outputs.

### ADF4351 Evaluation Board Analog Devices

The ADF4351- software Analog Devices was developed specifically for the ADF4351 evaluation board. The evaluation board has one pair outputs of the VCO and external clock a USB interface. Examine your schematics, you'll see that the CY7C68013A only used for the serial interface to the ADF4351. The CY7C68013A is a common chip with microcontroller and USB interface, so this chip is also used in the logic analyzers which has been written in the past in the Kunstmaan.



Figure 2 The evaluation board of Analog Devices. The ADF4351 is located at the bottom left. Top right is the Cypress CY7C68013A controller and USB interface.

During my search on the internet I came across a site of an amateur that evaluation board from Analog Devices has recreated with modules that are available on eBay. [4]

A module with Cypress CY7C68013A is already Euro 5 for sale on eBay and ADF4351-module for 20 Euro.

### **ADF4351 module eBay**

Find your eBay / AliExpress on ADF4351 you become engulfed with modules of around 20 Euros. I can not make any recommendations with whom you should buy. Also, I do not know who designed the original board. Of course, the documentation is missing.

The module, of course, contains the ADF4351 and a crystal to generate the clock. The 10-pin pinheader requires the digital control signals to be connected. Furthermore, there is also a connection for an external power supply. See also Figure 2.

According to the stories on the internet, you have to pay attention to a number of things:

- Are the SMA connectors well soldered? It is a 0.8mm thin PCB and sometimes used SMA connectors for printing 1.6mm. With me are the connectors on the bottom of the print out not soldered.
- Sometimes there is a ADF4350 on the module, these runs at 137.5 MHz (where we know the frequency?) To 4400 MHz. The print can still be ADF4351. The chip is then sand-blasted and printed again.

Furthermore, it is noticeable that the number of via's at the bottom side (GND) of the circuit board is minimized.



*Fig 3 the left, the two outputs, to the right the port for an (optional) clock. If you look closely you see that the SMA connector and not to the bottom soldered.*

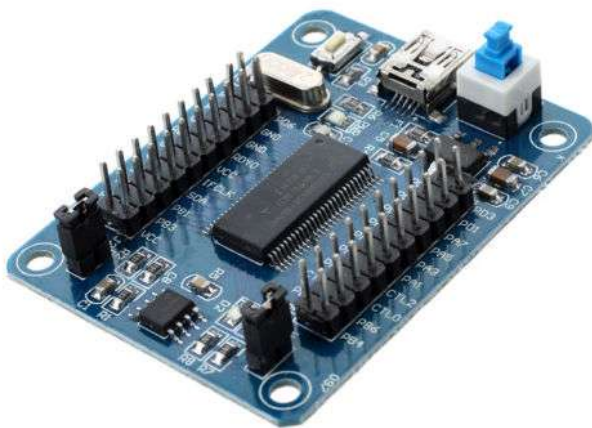
## Connecting

- The board has two outputs with SMA connectors. Use your only one? Then the other one with a 50 Ohm terminator.
- SMA connector for an external clock. This is optional, there is a 25 MHz oscillator present.
- Pinheader CLK. The clock signal from the microcontroller. The data is on the rising edge of the CLK signal. Connect to pin PA1 on CY7C68013A module.
- Pinheader DAT. The data signal to the microcontroller. Connecting pin PA2 on CY7C68013A module.
- Pinheader LE. With (L)oad (E)nable-signal of the shift register, the data is put through to the registers. Connecting pin PA0 on CY7C68013A module.
- GND. The GND to CY7C68013A module.
- Power supply of 5 - 9V

## CY7C68013A module of eBay

The board what I have, has an imprint of geeetech.com probably the designers. They have a nice wiki page with more information. [5]

Both jumpers must be fitted to the board.



*Fig 4 Upper right, the on / off button. Furthermore, all the CY7C68013A pins have been brought out. The 8-pin EEPROM is not used.*

## Software for the CY7C68013A

Then you need some software (you must have an account with Cypress)

Download "Cypress's EZ-USB development kit" [6], you will need it for the tools that we will use immediately. This allows us the hex file load the CY7C68013A, which let's think this is the official evaluation board.

Then we have (also, see [4] Linux) to recognize the USB drivers for the CY7C68013A Windows. The drivers can be downloaded from [7].

Connect CY7C68013A- module with a USB cable to the PC and connect the ADF4351-print to. I use an external power supply because I do not trust the USB voltage from my computer. The output of the ADF4351 with a 10dB attenuator is connected to the spectrum analyzer.



*Figure 5 The ADF4351- and through-connected to the CYC68013A modules and connected to the spectrum analyzer and the computer.*

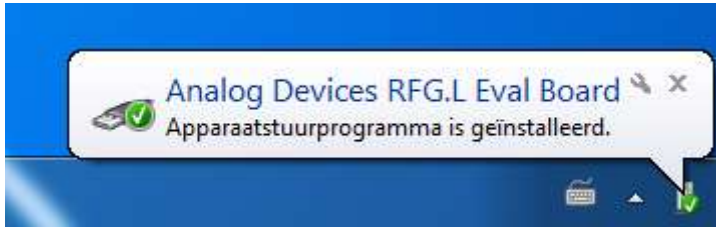
Then we need to install the software for the ADF435x [3].

And then the trick to "fool" the software. The program "Cypress USB Console" choose the menu "Options / EZ-USB Interface"

Press the Download button and select the file "adf4xxx\_usb\_fw\_2\_0.hex" that is in the ADF435x program directory. The hexfile is automatically uploaded.

## Analog Devices software for the ADF4351

You start the software and select ADF4351- ADF4351, and the USB board. Press Connect.



*Fig 6 Windows shows that a development board found.*

You get the following screen.



*Fig 7, the evaluation board is recognized.*

Then click on the Main Controls tab. Here you can set the frequency and the like. It takes a long way to discuss all options. On the site of Analog Devices can find all the information.

In the example below I entered 250 MHz. The output level is at -4dBm. Press "Write All Registers" (right bottom) and you see that everything is written to the ADF4351! First register 5, then 4 etc.

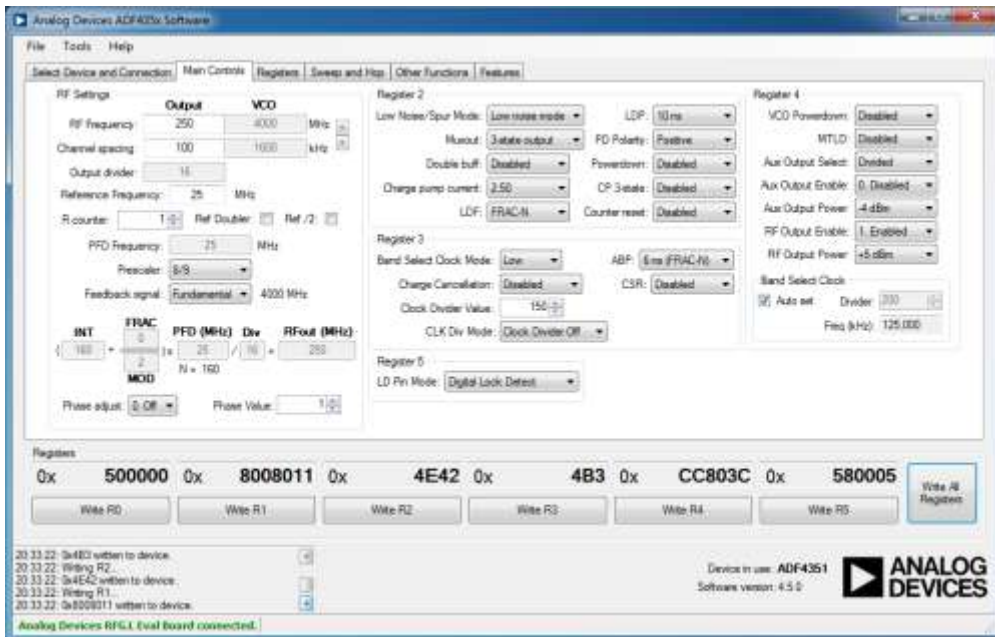


Fig 8 The frequency you enter in the output section. Then press Write All Registers.

The spectrum analyzer you will see the image below. The harmonics are strong because there is no filtering. Depending on the application, you will need to use filtering.

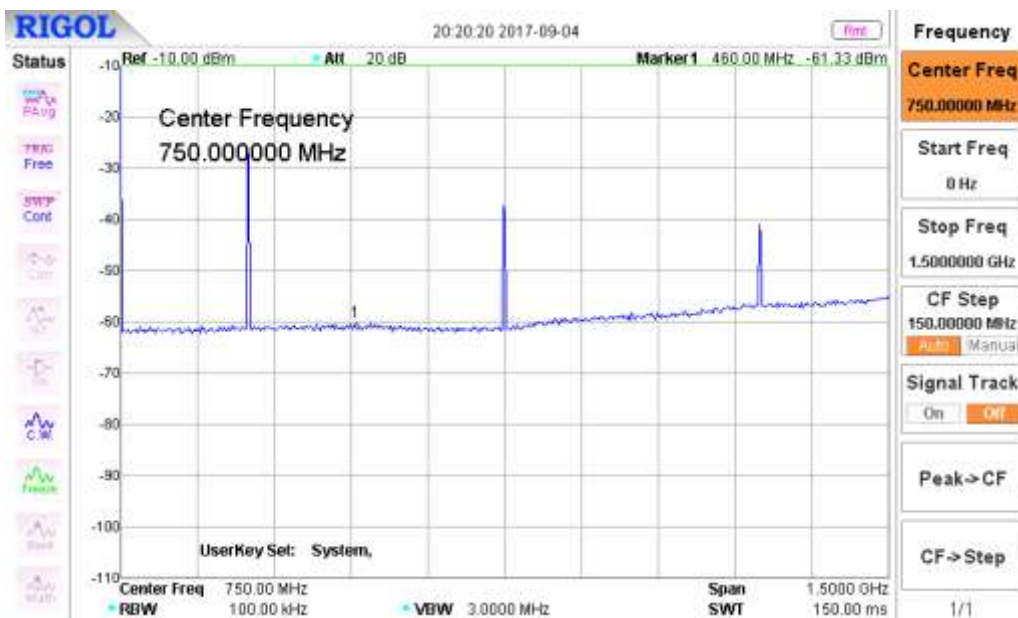


Fig 9 The result of the spectrum analyzer, the input is a 10dB attenuator. The output power looks lower then entered. Many harmonics, there is no filtering.

## Concluding remarks

A focus is the power consumption. The ADF4351 uses between the 110 and 144 mA (in the case 4400 to 35 MHz, output 5 dBm). At an output level of -4 dBm, the current consumption is 14 mA lower. The chip (and pcb) get hot!

With this type of single-chip solutions it is made very easy for the amateur radio to also experiment on the high frequencies. Maybe we'll encounter this chip again in future designs, who knows. This article attempted to get started with this chip!

It is of course nice to do this as an experiment, but if you seriously want to experiment with the ADF4351, then buy the evaluation board from Analog Devices. Of this you can be sure that it is properly designed. The eBay board is obviously a dud.

## Links

[1] Components of Mini-Circuits that are EOL

<https://ww2.minicircuits.com/pcn/PCN17-003b.pdf>

[2], Data Sheet of the ADF4351

<http://www.analog.com/media/en/technical-documentation/data-sheets/ADF4351.pdf>

[3] Evaluation Software for the ADF4351

[http://www.analog.com/media/en/evaluation-boards-kits/evaluation-software/ADF435x\\_v4\\_5\\_0.zip](http://www.analog.com/media/en/evaluation-boards-kits/evaluation-software/ADF435x_v4_5_0.zip)

[4] ADF4351 with a CY7C68013A

<https://www.youtube.com/watch?v=xJ0Mfuj0KUQ&t=2846s>

[5] on the Wiki CY7C68013A-board

<http://www.geeetech.com/wiki/index.php/CY7C68013>

[6] Development kit CY3684

<http://www.cypress.com/documentation/development-kitsboards/cy3684-ez-usb-fx2lp-development-kit>

[7] Drivers for the CY3684

<http://www.cypress.com/knowledge-base-article/drivers-ez-usb-fx1-fx2lp-and-fx3-kba94413>

Projects of amateurs

<http://www.kh-gps.de/adf4351.htm>

[http://pa0rwe.nl/?page\\_id=65](http://pa0rwe.nl/?page_id=65)

## **Report of the members meeting 9 September 2017.**

### **Opening by the chairman.**

Eumetsat will end next April with direct LRIT broadcasts from MSG.

Our sister organization GEO stops publishing their magazine; There is too little supply of items. A response by our member admin on the GEO mailing list has provided 5 new members for our workgroup.

Parking can be found free of charge next to the building; there is room for about 5 cars.

On November 4 we will be in Apeldoorn on the Radio Amateur Day. Job de Haas, Hendrik Jalving, Arne and Ben will staff the position.

On October 8th, there is an open day at Estec. Free, but you need to register.

### **Report ALV May 13, this was published in the June magazine**

There are no comments.

### **Setting the agenda**

No additions.

### **Management**

The website needs to be addressed. The website has been hacked; Job de Haas quickly got him back in the air. This indicates that a webmaster is needed.

### **Satellite status**

Given by Arne; See further in this magazine.

### **Anything else**

Harrie v. Deursen has been working hard on testing the new decoder and QPSK receiver. He has made a QPSK measuring transmitter; which is modulated with decoder / generator data. For example, the entire signal path can be tested: generator, transmitter, receiver, decoder.

Hendrik Jalving has made a new interdigital filter; loss in the passband is 1.42 dB. This filter is for 1.7 GHz, meant to set between helical and amplifier. This is to prevent the amplifier from being outraged by all kinds of noise signals.

Peter Kuiper has made the QPSK receiver. He compliments the designers; The whole is well thought out and easy to build.

Elmar Bögels inquired about the status of the down converter; how big is the print? Ben shows a prototype, components must be ordered. For the 5V we think of a USB power supply; cheap and well-available, but they must be reliable and short-circuit protected.

Job de Haas shows a working rotor. To isolate the “power”-part from the rest he used optocouplers. In addition, he has made an interdigital filter, using the knowledge of Jos Disselhorst, who gave a lecture the previous time.

Rob has been working hard with the HRPT / METOP / Fengyun decoder. One problem was always having to change VCO's adjustments as from satellite; HRPT, METOP and Fengyun all have a slightly different bit rate.

Elmar reports that there are small relays that you could switch between different trimmers to solve this problem. Rob reports that this problem has already been resolved in another way (see his article in this magazine).

## **Closure**

### **Lecture: adjusting the QPSK receiver**

There are 4 QPSK receivers made by various members, which are now being measured. Comments are noted to make any adjustments in design and / or print for the second series of orders.

Rob Alblas  
(Secretary AI)



# satellietstatus

Arne van Belle, status 23 september 2017

POLAIR	APT (MHz)	HRPT (MHz)	Overkomst
NOAA 15	137.620	1702.5	morning/afternoon, HRPT weak
NOAA 18	137.9125	1707.0	early morning/late afternoon
NOAA 19	137.100	1698.0	afternoon/night
FengYun 3A	none	1704.5	AHRPT 2.80 Msym/s
FengYun 3B	none	1704.5	AHRPT 2.80 Msym/s
FengYun 3C	none	1701.3	AHRPT 2.60 Msym/s
Metop-A	uit(137.100)	1701.3	LRPT/AHRPT 2.33 Msyms/s
Metop-B	none	1701.3	Only AHRPT 2.33 Msyms/s
METEOR M N1	off (137.100 LRPT)	1700.0	Black image
METEOR M N2	137.100 LRPT	1700.0	LRPT / MHRPT
NPP/JPSS-1	none	7.75-7.85 GHz X-band with 15Mbits/s !	

NOAA 15, 18 and 19 are the last satellites that broadcast APT.

The LRPT signal from METEOR M N2 can be received with an SDR dongle.  
See <https://groups.yahoo.com/neo/groups/GEO-Subscribers/files/Meteor/>

NPP (NPOESS Preparatory Project) and the to be launched JPSS-1 (NOAA-20) broadcast only on the X-band with 15 Mbit/s. A tracking 2.4 meter diameter dish is recommended ! <http://npp.gsfc.nasa.gov/sciencedocs/2015-06/JPSS-1SpacecraftHRDtoDBSRFICDRev-May302012-470-REF-00184.pdf>  
<https://directory.eoportal.org/web/eoportal/satellite-missions/j/jpss>

FengYun 3A, 3B and 3C broadcast AHRPT, only Ben Schellekens new QPSK receiver can handle these signals. This AHRPT is not entirely according to the standard so even a Metop AHRPT receiver is not suitable for the FY-3 series! FengYun 3C also has a different data rate than 3A and 3B. Harrie van Deursen and Rob Alblas are busy receiving and demodulating these satellites. Rob has expanded his GODIL decoder and is now able to demodulate HRPT, Meteor HRPT, METOP and FY3A / B and FY3C!

Sentinel-3A has been launched successfully, this long-awaited Envisat successor cannot be received directly, but images can be received via EUMETCast HVS-1.

GEOSTATIONAIR	APT (MHz)	(SDUS)/PDUS (MHz)	Baanpositie
MSG-4 (MET-11)	1691 LRIT	1695.15 HRIT	3.4 graden W, in testphase
MET-10	1691 LRIT	1695.15 HRIT	0 degrees W, operational
MET-9	1691 LRIT	1695.15 HRIT	9.5 degrees E, RSS
MET-8	no LRIT	-	3.5 degrees E, Back-up
GOES-E (no. 13)	1691 LRIT	1685,7 GVAR	75.0 degrees W via Eumetcast
GOES-W (no. 15)	1691 LRIT	1685,7 GVAR	135 degrees W via Eumetcast
GOES 14	1691 LRIT	1685,7 GVAR	105 degrees W, Backup
GOES 16			moves to position GOES- E in November
Elektro-L1	1691 LRIT	1693 HRIT	76 degrees E, via Eumetcast
MTSAT-1R	1691 LRIT	1691 HRIT	140 degrees E, Backup for MTSAT2
MTSAT-2	1691 LRIT	1687.1 HRIT	145 degrees E, via Eumetcast
Himawari-8	no LRIT	no HRIT	Operational, only via HimawariCast
Himawari-9			launched on 2-11-2016
FengYun 2D	-	-	86.5 degrees E
FengYun 2E	-	-	104 degrees E, now via Eumetcast
FengYun 2F	-	-	112.5 degrees E, Backup
FengYun 4A	-	-	86.5 degrees E, commisioning

MET-10 is now the operational satelllite.

EUMETSAT plans to stop the direct LRIT broadcasts on 1691 MHz next April!

[https://www.eumetsat.int/website/home/News/DAT\\_3247528.html?lang=EN](https://www.eumetsat.int/website/home/News/DAT_3247528.html?lang=EN)

We will miss this signal at 1691 MHz, it is a useful reference for receivers and rotors!

Dish owners be aware: From 9th to 15th of October the sun is right behind Eutelsat 10A between 12:59 and 13:09 Dutch local time.

The EUMETCast signal will disappear a couple of minutes. After 15 October, check that your LNB cap is not cracked or melted as a wet shiny dish can focus all sunrays on the LNB!

The 2nd EUMETCast transponder T2 was converted to full bandwidth on July 3 and shifted slightly to 11387.500 MHz Horizontal, The Symbol Rate and Mode is now the same as HVS-1 (33000 kSym/s DVB-S2, CCM mode, MODCOD 16APSK2/3). This transponder will only broadcast HVS-2 service.

Just like HVS-1, for good reception you can use a 90 cm dish (good weather) but you need a 120 cm for reception under rainy conditions.

<https://www.eumetsat.int/website/home/TechnicalBulletins/EUMETCast/index.html>

Manufacturers can now already receive live GOES 16 data on T2 / HVS-2.

Amateurs will likely be able to apply for this data in November.

Unfortunately, GOES 16 data is in netCDF format. Besides SNAP, Hugo van Ruys's EUMETCastView will supports this in the future.

<http://hvanruys.github.io/>

Since July, there have been signal strength variations on EUMETCast transponder T1 at 11263 MHz Horizontal.

Unexpectedly, Eutelsat, the satellite administrator allowed another service at 11263 MHz Vertical!

Due to "congestion" in the TV satellite world, we are seeing increasing numbers of transponders on Eutelsat 10A being used. For optimal signal

quality (SNR), you must rotate your LNB (Skew) to null out this vertical transmitter. A moderate or poor quality LNB can suddenly cause problems if it has a poor damping for the vertical signals. This is called cross-polarization isolation. This value should be better than 22 dB and determines how much the vertical signal is attenuated while the LNB receives horizontal.

But neighboring satellites such as 7 East and 13 East also use frequencies on or near 11263 MHz. A better LNB won't solve this, dishes smaller than 120 cm have a larger opening angle and will experience more disturbance from neighboring satellites.

On my dish, I experimented with silver-colored CD-ROMs fixed to the outer edge with a magnet (first on the east side and then on the west side). This increases the dish area slightly and makes it less sensitive to interfering signals from neighboring satellites. I was able to completely compensate for the sudden signal loss of 0.5 dB SNR. The row on the west side gave the best result, the row on the east side did not improve SNR. Unfortunately, this setup does not withstand a storm!

Meanwhile, the T1 transponder has been more balanced with the vertical transponder and the signal strength is back to the old value of 11.9 to 12.1 dB SNR on my WaveFrontier T90 without the CD-ROMs. HVS-2 on T2, however, remains approximately 0.4dB weaker.

EUMETSAT recommends repeating the fine alignment of your dish annually and double check for correct Skew of LNB. If possible, also check the focus (slide it in and out towards the dish).

If the old SNR values are no longer achievable, it may be necessary to replace the LNB with one that has better "Cross-polarization Isolation".

See "EUMETCast Europe Link Margins Explained" at

[https://www.eumetsat.int/website/home/TechnicalBulletins/EUMETCast/DAT\\_3569223.html](https://www.eumetsat.int/website/home/TechnicalBulletins/EUMETCast/DAT_3569223.html)

#### Launches:

FengYun 3D	end 2017 ?
MetOp-C	launch in October 2018
JPSS-1	(NOAA 20 after commissioning) has been postponed to November 10, 2017
Electro-L3	end 2017?

EUMETCast can only be received in DVB-S2 VCM on Eutelsat 10A 10 degrees east since December 31, 2014!

To enable further growth in the amount of data transmitted via EUMETCast in the future, EUMETSAT migrated from DVB-S to DVB-S2 with VCM mode in August 2014.

The new transponder is on Eutelsat 10A, 11263 MHz Horizontal at 10 degrees East.

The broadcast standard is DVB-S2 VCM 8PSK 3/5 (Basic Service) or 16APSK 2/3 (High Volume Service) with a symbol rate of 33 Msps.

Unfortunately, DVB-S and most "DVB-S2 without VCM" receivers can no longer be used. With a special driver, some recent DVB-S2 receivers can be made suitable for Basic Service Only. (only the TBS-5980 and Skystar 2 eXpress HD, unfortunately, this does not apply to the Skystar HD USB box)

The signal at 10 degrees east has a larger bandwidth and is therefore weaker than before.

Recommended dish diameter is 80-90 cm for Basic Service and at least 120 cm for High Volume Service 1 and 2  
During heavy rain showers, the signal may disappear earlier and longer.

Since February, EUMETSAT is using a second transponder, T2 at 11387.5 MHz Horizontal for HVS-2.

[https://www.eumetsat.int/website/home/TechnicalBulletins/EUMETCast/DAT\\_3340783.html](https://www.eumetsat.int/website/home/TechnicalBulletins/EUMETCast/DAT_3340783.html)

Because T2 is on the same band and polarization (Horizontal), you can receive it with the same dish, LNB and cable. Until recently, only test (dummy) data was received on HVS-2.

In mid-November, live GOES 16 data will be broadcasted on HVS-2.

Unfortunately, the SR1 receiver cannot receive both transponders at the same time. The multi-tuner cards like TBS6983 and 6903 can receive both Transponders simultaneously. On the PC you will need to start Tellicast 3 times with different configurations. Receiving T1 and T2 gives a lot of data (27 Mbps Basic Service, 34 Mbps HVS-1 and 77 Mbps HVS-2). To do this, you need a ramdisk and multiple disk drives to decode all data.

The reception of EUMETCast data is free of charge for amateurs, but you must register with EUMETSAT. You have to buy software (60 Euro) and key (40 Euro) once.

On the EO Portal, EUMETCast allows users to log in and view their personal information and settings. Registration as a new user and renewal of the license is also possible here. You can add services or products to your ECU on-line.

A new improved Tellicast Client v2.14.2 is send out via "Info-Channel-1".

This new client is only required if you want to receive HVS-1 or HVS-2 data.

Thanks to David Taylor and Douglas Deans for the info.

Many frequencies can be found here:

[http://mdkenny.customer.netspace.net.au/metsat\\_frequencies.html](http://mdkenny.customer.netspace.net.au/metsat_frequencies.html)