STAR-XL: Student Transponder for Satellite Ranging on X & L-band

C. P. Bridges[1], P. Bladen, S. Lane, P. Hope, M. Friesch, T. Brown[2], D. Bowman, G. Shirville[3]

Abstract

The ESA ESEO Mission [1] included an amateur radio payload [2]. The results of which included the development of radio technologies that utilised final year student projects over a 5 year period. Many lessons regarding compliance and process enabled a new payload to follow: the Student Ranging Transponder Radio for X-band and L-band (or STAR-XL). The STAR-XL design leverages key aspects of the ESEO payload design for a generic CubeSat platform; including TT&C voltage and current sense circuitry, receiver circuitry, and flight software. But instead of a maximum 4800 bps telemetry and transponder system - the STAR-XL targets a 100 kHz bandwidth system that will allow faster downlink rates that are forward error correction, link margin and modulation order dependent. With 100 kHz bandwidth, the linear receiver is designed to also operate as a transponder - enabling ranging and navigation applications such as orbit determination and further experiments from amateur radio groundstations. This paper details the recent student project efforts in three key areas: a new STM32-based on-board computer, an X-band up-converter board and dual X/L band patch (as shown in Fig. 1). The new OBC includes an IQ modulator for transmitting complex waveforms and an optimised flight software suite that takes advantage of dual DMA hardware on-chip to reduce overheads. The X-band upconverter board required the development of new safety interlock and RF chain circuitry on a Rogers (RO4350B) PCB material. A new dual X/L-band patch antenna and filter circuit is also built and measured. Each of these projects has led to new lessons and increased the real-world case studies used to teach spacecraft avionics.
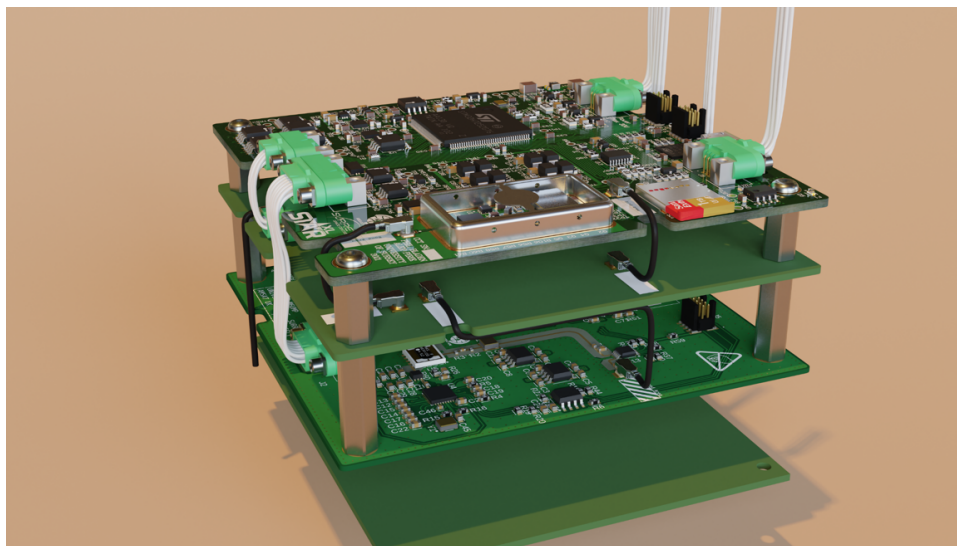
Figure 1. STAR-XL CAD Blender Model

Keywords

Transponder, Satellite, Ranging, Radio

[1] Corresponding author: Surrey Space Centre, United Kingdom, c.p.bridges@surrey.ac.uk
[2] 5G/6G Innovation Centre, Uni. of Surrey, United Kingdom, t.brown@surrey.ac.uk
[3] AMSAT-UK, United Kingdom, G3VZV@amsat.org

1. Background

A 1U CubeSat payload, known initially as FUNcube-NEXT and later STAR-XL, is being developed based on AMSAT-UK's FUNcube-4 payload, which is on the European Student Earth Orbiter (ESEO) satellite that was launched in December 2018 by ESA. FUNcube-NEXT will comprise a flight computer board that will be an improved version of the one in FUNcube-4. It will support downlink modulation and bandwidth experiments with the aim of increasing the downlink data rate. It will also have a new microcontroller both to support the higher downlink data rates and because the part used in FUNcube-4 has reached end-of-life status.

2. Flight Computer Development

When developing this flight computer board, a standalone Elegant Bread Board (EBB) flight computer was developed. It has all the functionality of the final flight computer board, including an RF oscillator and RF IQ modulator for the experimental downlink improvements, and in addition includes facilities to enable two distinct candidate microcontrollers (a low power low performance one (STM32L4P5) and a high-power high performance one (STM32H725)) to be evaluated in reality. These facilities include the ability to switch between the candidate microcontrollers, instrumentation to enable the power consumption of key parts of the design to be analysed, and USB communication between the microcontrollers and a PC.

The RF results include PSK constellation diagrams resulting from configuring the development board to produce a variety of PSK modulations. New dynamic MCU clocking results show that the MCU can change its clock frequency dynamically, how the MCU utilisation, current consumption and power consumption depend on the clock frequency, and also how these are affected by enabling the MCU caches.

Power consumption analysis results show that the STM32H725 easily meets the MCU power budget requirements when clocked at 256 MHz despite being a high power MCU and that higher clock speeds are possible provided certain precautions are taken. At 256 MHz, its performance is approximately 12 times that of the processor used in the FUNcube-4.

In the expected orbit for FUNcube-NEXT, the radiation exposure should not be significantly greater than on the surface of Earth, except when over the SAA. As a result, using radiation-hardened components is deemed unnecessary. However, SEUs must be tolerated, so hardware ECC availability will be a criterion for choosing MCUs. Should an MCU without ECC be chosen, memory scrubbing techniques will be employed. Ultimately, should the STM32H7 be used as the flight computer MCU, the risk of memory corruption due to SEUs would likely be decreased to an acceptable operating level, even without additional firmware to perform memory scrubbing.

Several design aspects from the ESEO CCT [2] are to be carried forward to the flight computer development board. The existing current sensing approach, including chosen current sensor, input filter and ADC will be incorporated into the new design. The CAN bus architecture and transceivers will also be kept.

The Analog Devices design to couple a single-ended DAC to the IQ modulator is adapted to fit the project objectives and includes a 5th order Bessel filter to remove DAC quantization noise.

Finally, established PCB layout techniques will be employed whilst designing the PCB for manufacture. A single ground plane will be used to avoid unintentional ground loops, and the analogue and digital domain will have separate power planes.
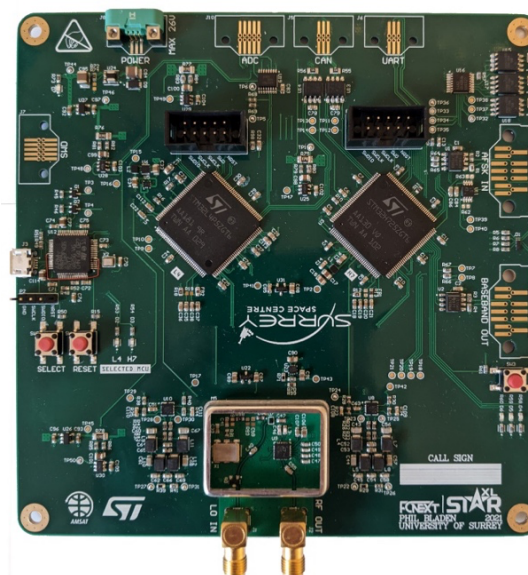

Figure 2. STAR-XL Flight Computer EBB PCB

2.1. Dynamic MCU Clocking

To test the STM32H7 dynamic MCU clocking, firmware was written that gave the MCU a fixed amount of work to do per second, while the MCU core clock was cycled through a series of frequencies and the MCU utilization, current

and power were measured. The firmware used the FreeRTOS real-time operating system, with the work task running at the highest priority.

Percepio Tracealyzer [3], a commercial instrumentation tool for FreeRTOS firmware, was used to monitor the MCU utilization and display it on the PC. Tracealyzer has two modes of operation. In one mode, known as snapshot, the instrumentation data is stored on the MCU during logging, and then downloaded at the end. In the other mode, the instrumentation data is streamed continuously to the PC in real-time. The real-time streaming was found to work with the STM32F767 on an STM32 Nucleo board, and later on an STM32H743 on an STM32 Nucleo board after working with Percepio and receiving an update to the software. The dynamic clocking experiments were conducted using the non-real-time instrumentation (snapshot) mode.
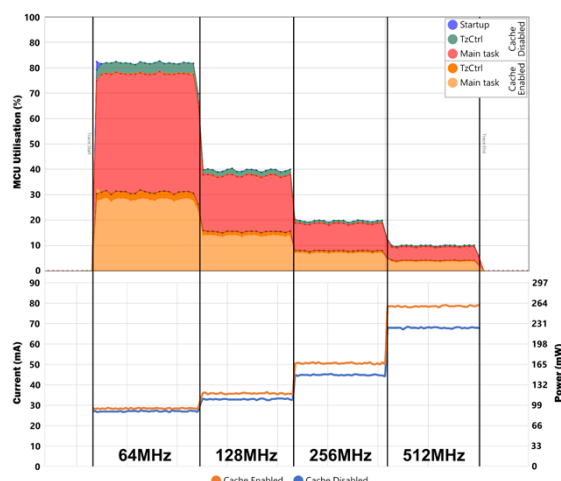

Figure 3. Percepio Tracealyzer showing Frequency vs Current Consumption.

Note that enabling the caches leads to a performance increase of approximately 167% for only a small increase in power consumption.

For use in Space, it is often preferential to disable caches as they are particularly vulnerable to SEUs. However, in this case, the increase in performance is significant enough to consider alternative approaches to handling SEUs vulnerability.

The new flight computer can reliably generate BPSK signals up to 19200Baud, a considerable improvement over ESEO, and has been tested up to 115200Baud, of which the hardware is fully capable but limited by single-buffered-DMAs. AFSK signals can be decoded in the same manner as ESEO, however optimisations have been performed to reduce the CPU usage from the 50% on ESEO, to 37% as a baseline

with the microcontroller, and finally further optimisations were made to optimise the entire system down to 13% CPU usage. This gives a lot of room for future expansion.

The new flight computer also operates at a much lower power consumption than ESEO, operating at 94.1 mW with all systems including the redundant CAN interfaces running at full capacity, all while maintaining redundant oscillators and backup power supplies for adverse situation recovery.

3. Flight Software

During the development of the STAR-XL hardware, a new fork of flight software was being written. The functionality encompassed by this software includes the sampling and decoding of the AFSK uplink baseband, the gathering & encoding of data from the payload and the generation of the BPSK downlink baseband. This was all handled by the new STM32H7 series CPU. A key feature of the software is the requirement for multiple functional areas of the program to be executed in a pseudo- parallel way; this necessitated the use of a scheduler to separate the execution time of the processor into threads.

FreeRTOS was reselected for this purpose due to its low memory impact, high performance and free use license. FreeRTOS also has excellent existing integration with the STM32 series microcontrollers, having been rigorously tested as part of the STM32CubeMX/IDE development tools provided by STMicroelectronics.

The decision to treat the flight software as a completely thread-parallel system has introduced the requirement for resource-locking, however this extra performance overhead is offset by the improvements in single core performance of the CPU and the inherent capability of porting to a dual (or greater, in the future) core CPU with minimal extra software effort required.

3.1. Software Architecture Improvements

As seen in thread execution time in the ESEO flight software, collisions can be seen between the uplink and downlink ISRs. While this has been partially resolved by reducing the processing time spent in each ISR and ensuring that nested vector interrupts are enabled, this has never solved the issue.

ESEO literature defines a "sampling jitter of only 1.8% of the sampling frequency in 1.2kbps mode and 7.8% in 4.8kbps [is observed]". Due to the low sampling frequency and the comparatively high clock speed of the

microcontroller this jitter should be a lot lower. It is obvious that the interrupts are still colliding, just in a considerably less noticeable and catastrophic way. Having 2 interrupts which are both highly time critical, and where there is a high probability of both firing at the same time, or while the other ISR is still running is a significant processor time contention issue. This is a perfect situation to leverage the high flexibility DMAs available on the new STM32 platform for both ADC and DAC sampling, in order to move the sampling load from the CPU to hardware and remove the reliance on hard-real-time interrupts.

In the new solution for STAR-XL, the DAC and ADC are configured to trigger on the rising edge of their respective hardware timer (internal to the CPU), the ADC will then convert a sample and trigger the DMA once it is complete. The DMA engine will then move the sample to a ring buffer in memory ready for the CPU to process. Likewise for the DAC, a hardware timer will trigger the DMA to move a sample from a memory buffer to the DAC, the DAC will then trigger and the sample will be converted to the output.

This approach has the benefit of eliminating the jitter caused by ISR collision, as the peripherals and DMA operate entirely independently of the CPU. From the perspective of the CPU core, a ring buffer is asynchronously filled with samples at a regular interval. The buffers are sized appropriately such that they will never be fully filled nor emptied by the DMA and peripherals; it is then the responsibility of the CPU to ensure that these buffers are adequately serviced. During this entire process there are minimal system interrupts, the scheduler is free to allocate CPU time as it sees fit.

The Uplink and Downlink are now entirely decoupled from the processes which generate their samples; the buffers now are able to be generated in an environment requiring considerably less hard-real-time performance. This also has the advantage that the CPU cycles that would otherwise have been used to trigger (and wait for) the ADC and DAC can now be used to prepare or process samples. This allows for a much higher sample rate to be used, which contributes to improving the probability of receiving accurate symbols from the uplink and can reduce the filtering requirements of the downlink. A higher temporal resolution can be developed at the DAC output. This allows an in-situ raised-root-cosine filter to be implemented which in turn allows a higher data rate to be achieved on the downlink without introducing harmonics.

The processor chosen should has a fully featured DMA engine along with 2+ independent ADC peripherals. Therefore, one ADC can be dedicated to reading the AFSK uplink using the DMA. To expedite these transfers and reduce bus contention the processor chosen should have a suitable inter-peripheral crossbar matrix (or manufacturer equivalent) to allow the DMAs to access RAM at high speed without impacting the speed or latency at which the CPU can access that same RAM. If there is a section of RAM that can be accessed using a different bus that would be ideal to allow for multiple simultaneous RAM operations to take place.

3.2. Profiling, Logging and Debug

Critical to debugging, testing and validating the software is a versatile and well-structured logging system. The ability to output logs to a debug UART or ITM is critical to debugging running software without interrupting it with breakpoints; even a log as simple as a heartbeat can be useful for determining if a processor is still running. Logs are also useful when performing tests to determine the number of packets transmitted and received from various communications interfaces, this is useful when performing statistical analysis of communications reliability.

The logging system is designed to be as low-impact as possible, while also maintaining high levels of functionality. A "Logger" is created for each file with source code that will be instrumented for logging, likewise the various "Loggables" for said Logger are also defined using macros of a similar structure. These Loggables include:

- Events – Log the hit of a particular line of code, such as a conditional or state.
- Function – Log the entry and return from a function, along with its execution time and other parameters
- Housekeeping Result – Special event type for the housekeeping thread to log a failure or success of a housekeeping activity.
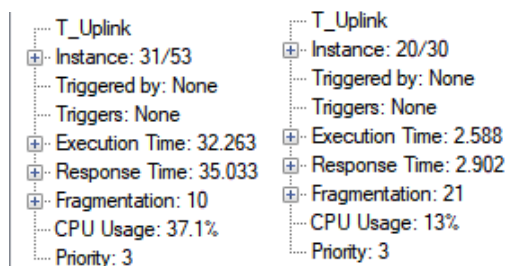

Figure 4. AFSK Uplink: Optimised Implementation CPU Usage Breakdown

3.3. Thermal Chamber Testing

The chamber was programmed with a thermal profile depicted in Fig. 5, in which the chamber and board was heated to +55C as fast as the chamber can safely allow, dwelled for a period then cooled as rapidly as the chamber will safely allow to -25C, another dwell and the cycle was repeated 4 times for a total of 4 hot dwells and 4 cold dwells. At the first hot and cold cycle a set of instantaneous state and statistical analysis tests were performed to obtain a baseline of the boards performance at the extremes of temperature expected during spaceflight. The cycles were left running overnight with the board depowered. As seen in the thermal data figure, there was a gap in the thermal measurement data caused by the university IT department restarting all networked computers at 3AM. Suitable provision should be made to not allow this to happen in future tests.
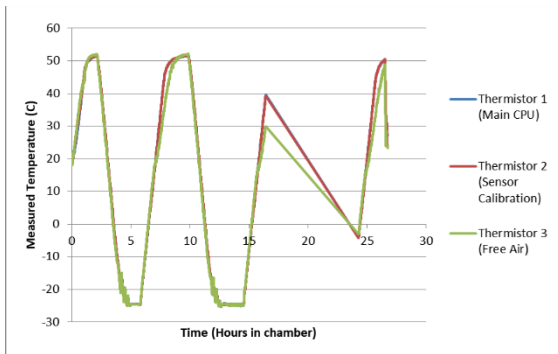

Figure 5. Thermal Chamber Internal Setup & Testing Recorded Thermistor Data

4. RF X-band Tx Board & Antenna

Iterative designs of the proposed RF chain for the STAR XL payload are shown below.
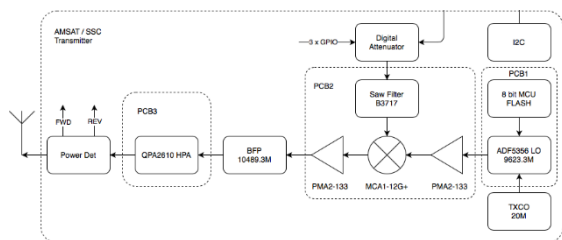

Figure 6. transmitter chain from the STAR XL system diagram

The received 1264 MHz signal arrives at the L-Band receiver developed and is mixed down to an intermediate frequency of 45 MHz. From here, the signals are separated, with the AFSK modulated commands being forwarded on to the On- Board Computer (OBC). The RF switch allows the transmitter to select whether to broadcast a response to the AFSK commands, or forward on the other signals that were not sent to the OBC. The signal then continues through the chain, where it is mixed up to another intermediate frequency of 866 MHz.

The intermediate 866 MHz signals are passed from the L-Band receiver section to the X- Band transmitter section. The intermediate frequency is mixed with a local oscillator set at 9623.3 MHz, producing a desired RF output at 10489.3 MHz, and the other modulation product at 8757.3 MHz. The transmitter chain then filters out the unwanted product and any of the local oscillator that has made it through to the output, where it is boosted by a high-power amplifier (PA) and then transmitted back to Earth.

A number of EBBs were developed including a high power amplifier and mixer / filter circuits in the following figures.
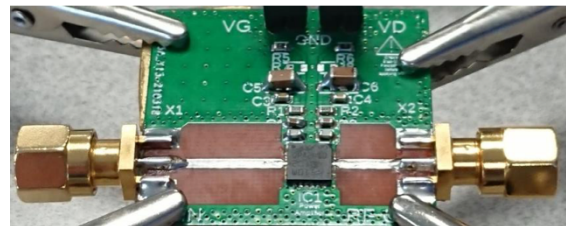

Figure 7. The modified HPA PCB, showing the new SMA connectors (with 50-ohm terminators attached)
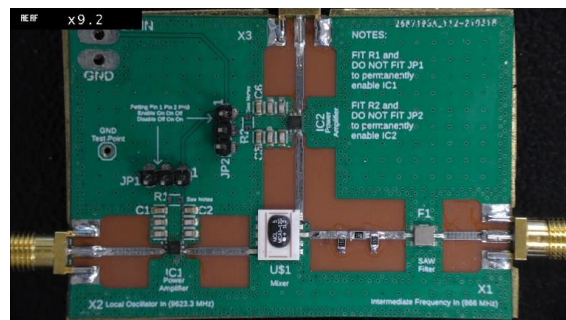

Figure 8. The completed circuit board for the RF Mixer, with modified SMA connectors and added Pi attenuator

We have verified that the DC response of the PMA2-133LN+ amplifiers matches that described in the datasheet. The MCA1-12G+ mixer performed as expected, generating a stable, low-noise signal at 10489.3 MHz as desired. The S-Parameters of the system were measured and found to be acceptable for the

performance of the system. Overall, the mixer EBB design performed as desired, and the parts selected for use in the block diagram are satisfactory for use in the final transmitter design. It may be necessary to add matching components at the outputs of the PMA2-133LN+ amplifiers, and the Pi attenuator values may be amended to provide different attenuation and better matching at 50Ω.

The High-Power Amplifier board had the most issues, with PCB design flaws, transmission line errors and supply rail anomalies. More than 60% of the power in the system was being wasted as heat. Despite this, the transfer response of the device followed the expected performance indicating a capable device if all these problems are rectified. After correcting the design, a final board design is developed.
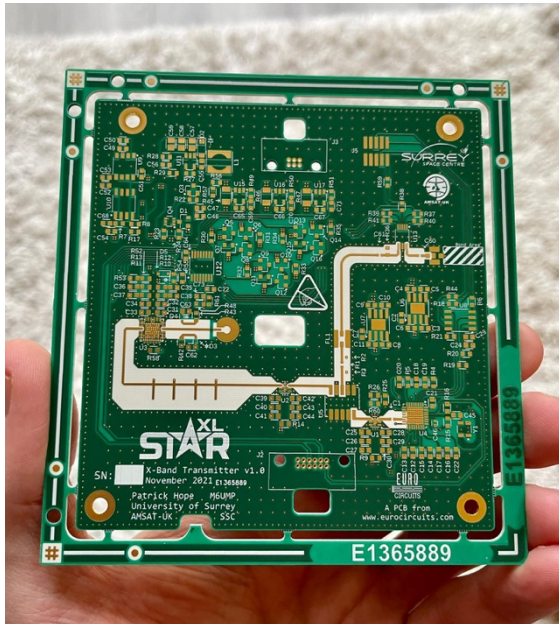

Figure 9. STAR-XL X-band Transmitter

Given the new X-band transmitter, a new antenna is needed for dual L-band receive and X-band transmit.

Table 1. Dual-band Antenna Design Parameters

Mechanical Properties	Dimension	
Overall dimensions	100.5x82.6x2.2mm	
Radius of Void in L-band patch R_{Void}	8.5mm	
Width L-band patch W_{Patch}	61mm	
Diagonals L-band patch L_{DLong}, L_{DShort}	92.5mm, 89.5mm	
Width of tuners in L-band W_{Tuner}	5mm	
Feed inset L-band x_{feed}	21.5mm	
Substrates & height h	1.524mm RO3003, Patch PCB	0.508mm RO4350B, Feed PCB
Aperture feeding dimensions X-band	See Table 12	
Tuner Dimensions X-band	See Table 13	
RF Properties	**L-band**	**X-band**
Centre frequency f_r	1.264GHz	10.4893GHz
Bandwidth	18.2MHz (S11 ≤ -10dB)	919MHz (S11 ≤ -15dB)
Axial Ratio at f_r and in direction of propagation (z-Axis)	1dB, AR BW 3.6MHz	2.2dB, 3dB AR BW 297MHz
Half Power Beamwidth (HPBW)	95°	106°
10dB Beamwidth	180°	160°
Power reduction at 120° beamwidth	4.8dB	4.3dB
Antenna Peak Gain	5dBic	4.9dBic
Polarisation	RHCP	LHCP
Cross coupling X-band to L-band	< -40dB	

Figure 10. X-band Antenna

Here, the measurement results of the manufactured combined antenna are presented. First, the L- band measurements are investigated and then the X-band measurements are also shown. Figure 10 shows images of the assembled combined antenna as seen from the top side (a) with the L- and X- band patches.

Although the bandwidth is reduced in the aperture integrated antenna, the design can still be used well given the possibility of tuning. Despite this limitation, the simulated design fulfils all design requirements of this work can thus be manufactured.

Acknowledgements

References

[1] Bruzzi, D., Tortora, P., Giulietti, F., & Galeone, P. (2013). European student earth orbiter: ESA's educational microsatellite program.

[2] Bartram, P., Bridges, C. P., Bowman, D., & Shirville, G. (2017, March). Software defined radio baseband processing for ESA ESEO mission. In *2017 IEEE Aerospace Conference* (pp. 1-9). IEEE.

[3] Percepio Tracealyzer, https://percepio.com/tracealyzer/