Heisenbug

Тестируем мобильный телефон по-другому. Полное погружение в HW стенды

Докладчик: Давыдова Надежда

kaspersky

Кто рассказывает?

Надежда Давыдова
Nadezhda.Davydova@kaspersky.com

kaspersky

О чём пойдёт речь

- Зачем нам Continuous Integration (CI)
- Новая ОС и области её применения
- Тестирование в эмуляторе и почему этого недостаточно
- Тестирование с применением оборудования

kaspersky

Зачем нам CI

- Автоматизированный регресс
- Повторяемость
- Стабильность тестов
- Расширение тестового покрытия
- Экономия времени на документировании

KasperskyOS

- Собственная разработка
- Микроядро
- Интеграция с модулем безопасности

Области применения

Kaspersky
IoT Infrastructure Security

Kaspersky
Secure Remote Workspace

Kaspersky
Automotive Adaptive Platform

kaspersky

Тестирование мобильного телефона на KasperskyOS

kaspersky

Тестирование в эмуляторе

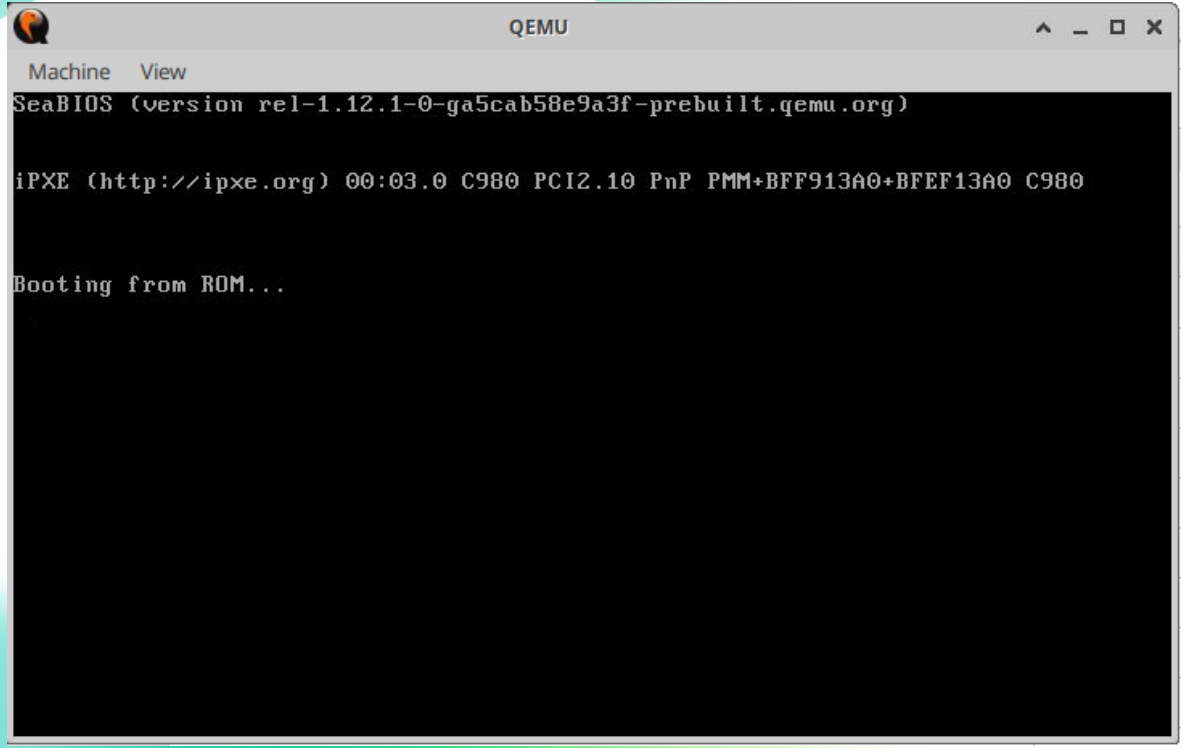
- Что тестируют обычно?
- Почему этого недостаточно?

Что тестируют обычно?

- Работу приложений и сервисов в ОС
- Взаимодействие компонентов

Эмулятор

Эмулятор архитектуры с открытым исходным кодом



kaspersky

Преимущества тестирования в эмуляторе

- Легко интегрировать в CI
- Высокая повторяемость тестов
- Дешевле тестирования на устройстве
- Подходит для различных устройств

Continuous Integration

- Bash скрипты
- Библиотека QMP
- pytest, Python

Пример запуска

```
qemu-system-x86_64 \
-m 1024 \
-cpu core2duo \
-nographic \
-netdev tap,id=net0,ifname=tap0,\
            script=no,downscript=no
-kernel ${PATH}/kos-qemu-image
```

Управление эмулятором

```
- qmp tcp:localhost:4444,server,nowait
```

```
File  Edit  View  Search  Terminal  Help
burenkov_o@burenkov-vb:~$ ./QEMU_RUN.sh
RTNETLINK answers: Operation not permitted
W: /etc/qemu-ifup: no bridge for guest interface found
RTNETLINK answers: Operation not permitted
W: /etc/qemu-ifup: no bridge for guest interface found
qemu-system-x86_64: multiboot knows VBE. we don't
```

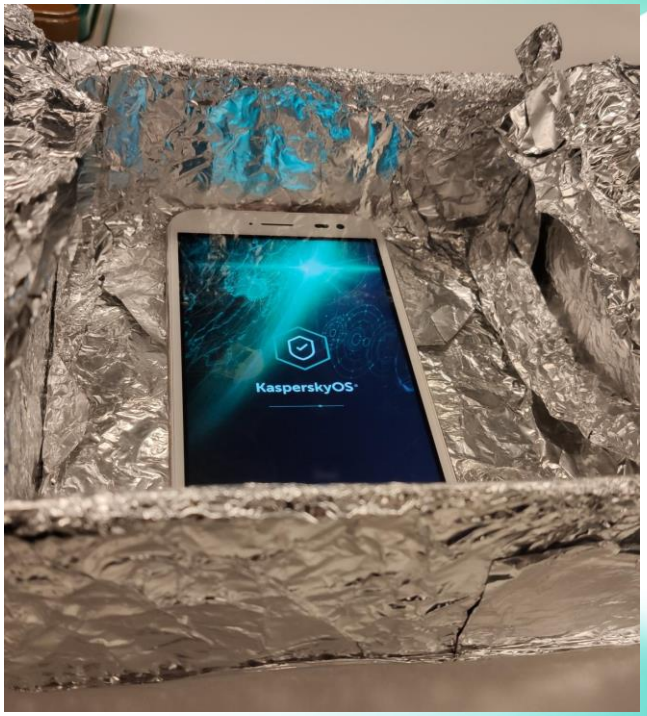
Почему этого недостаточно?

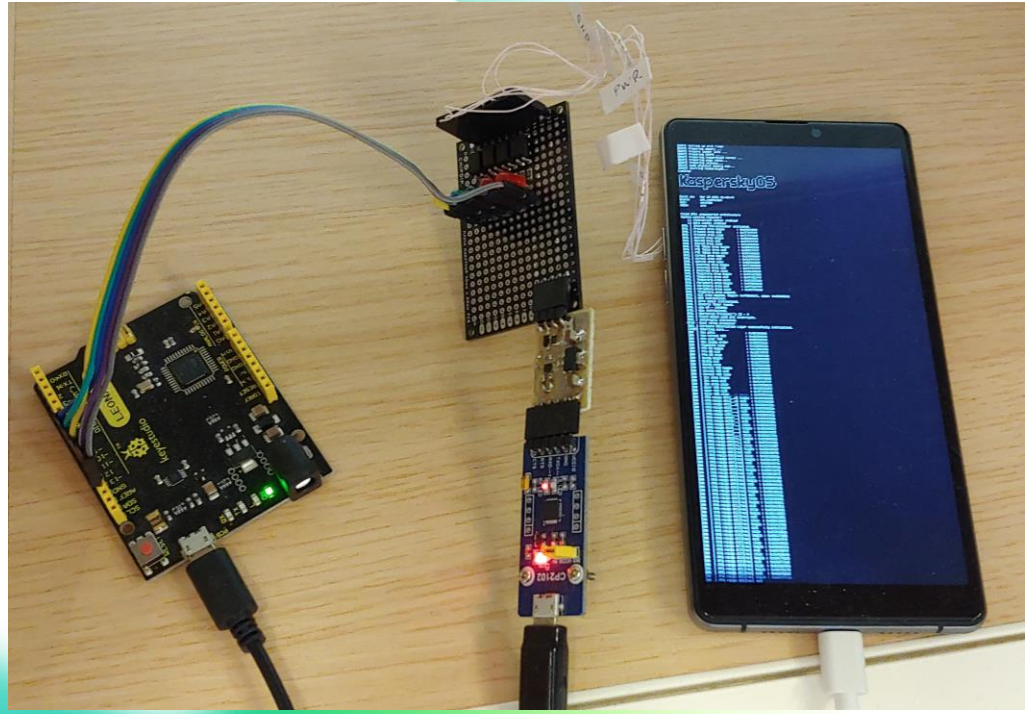
- Трудоёмко писать заглушки реальных устройств
- Не позволяет обнаружить ошибки в работе драйверов
- Телефон может работать по-другому
- Мы тестируем ОС

Что нужно протестировать

- Регистрация в сотовой сети
- Звонки и сообщения
- Передача данных
- Энергопотребление
- Работа периферийных устройств

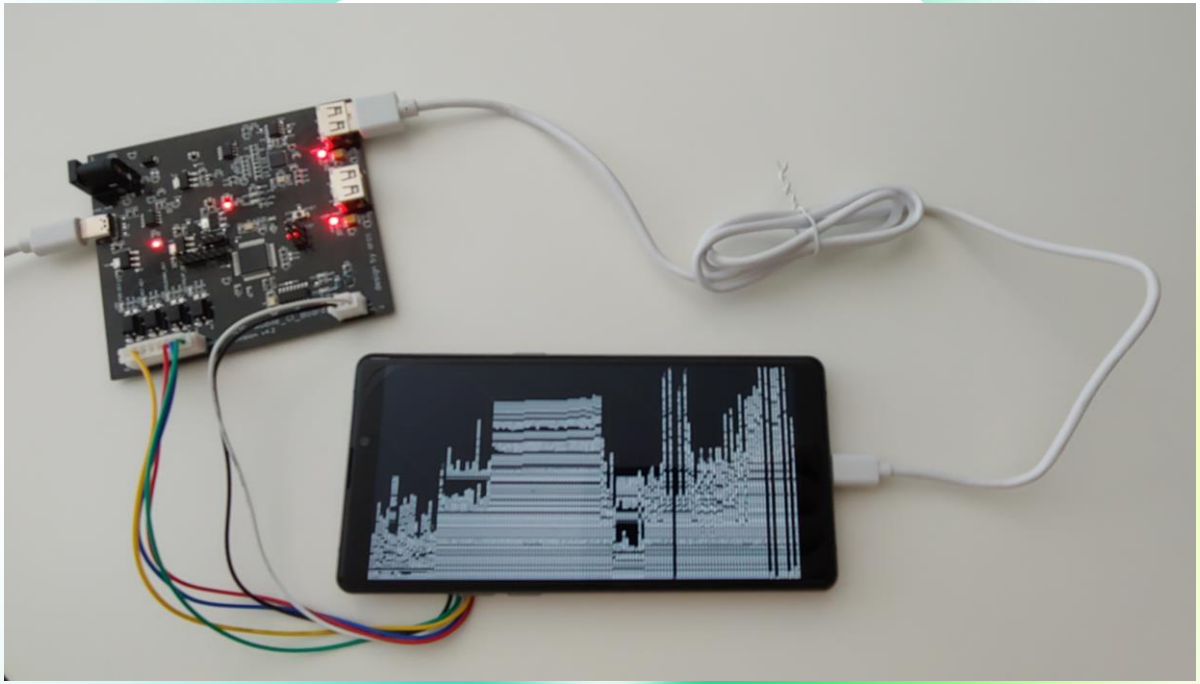
Первые опыты



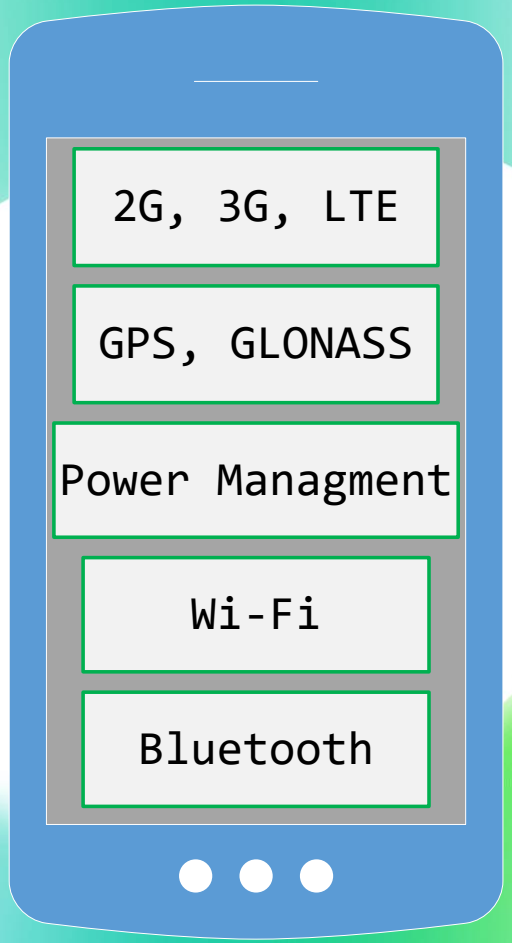


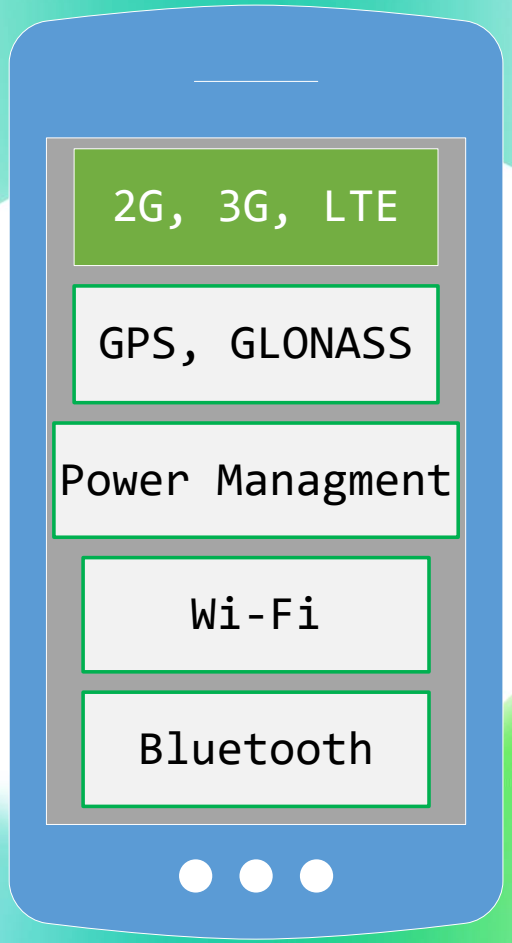
kaspersky

Прототипирование



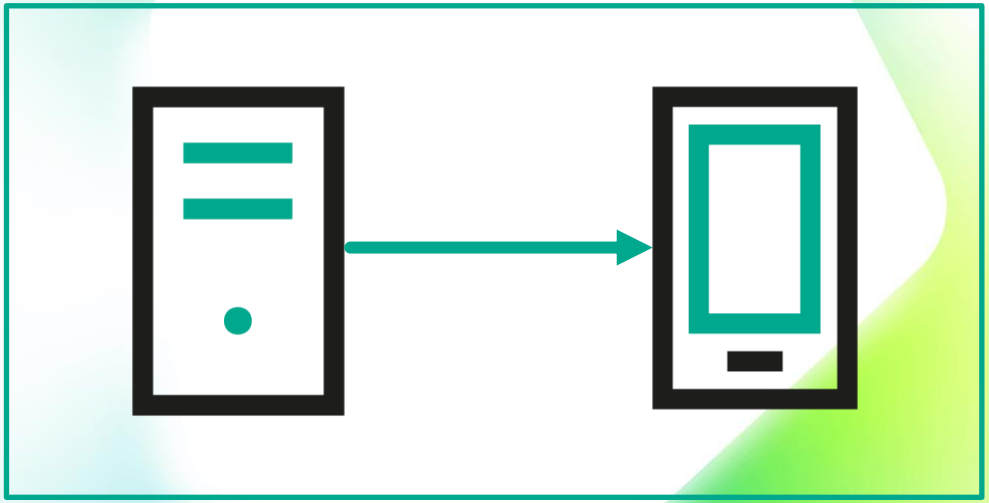
kaspersky


2G, 3G, LTE
GPS, GLONASS
Power Managment
Wi-Fi
Bluetooth

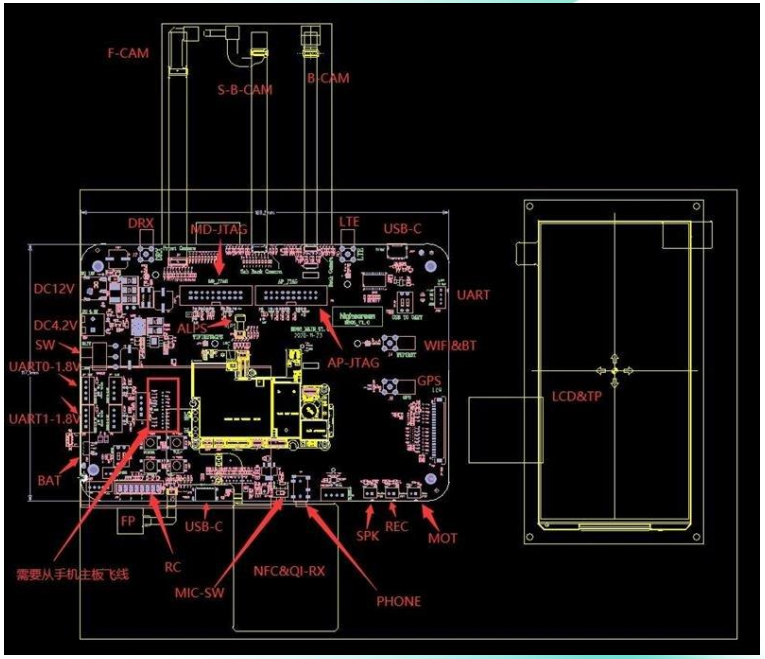

2G, 3G, LTE
GPS, GLONASS
Power Managment
Wi-Fi
Bluetooth

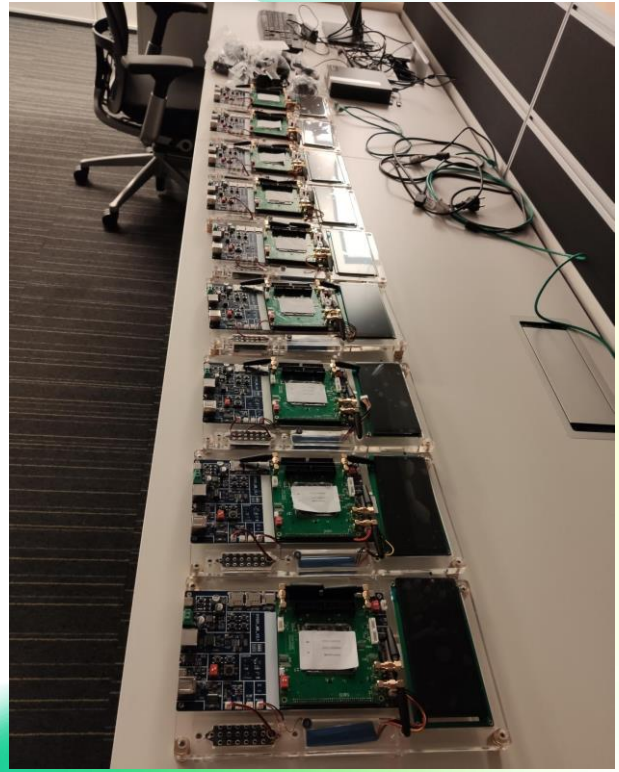
Как доставить сигнал?

Как доставить сигнал?

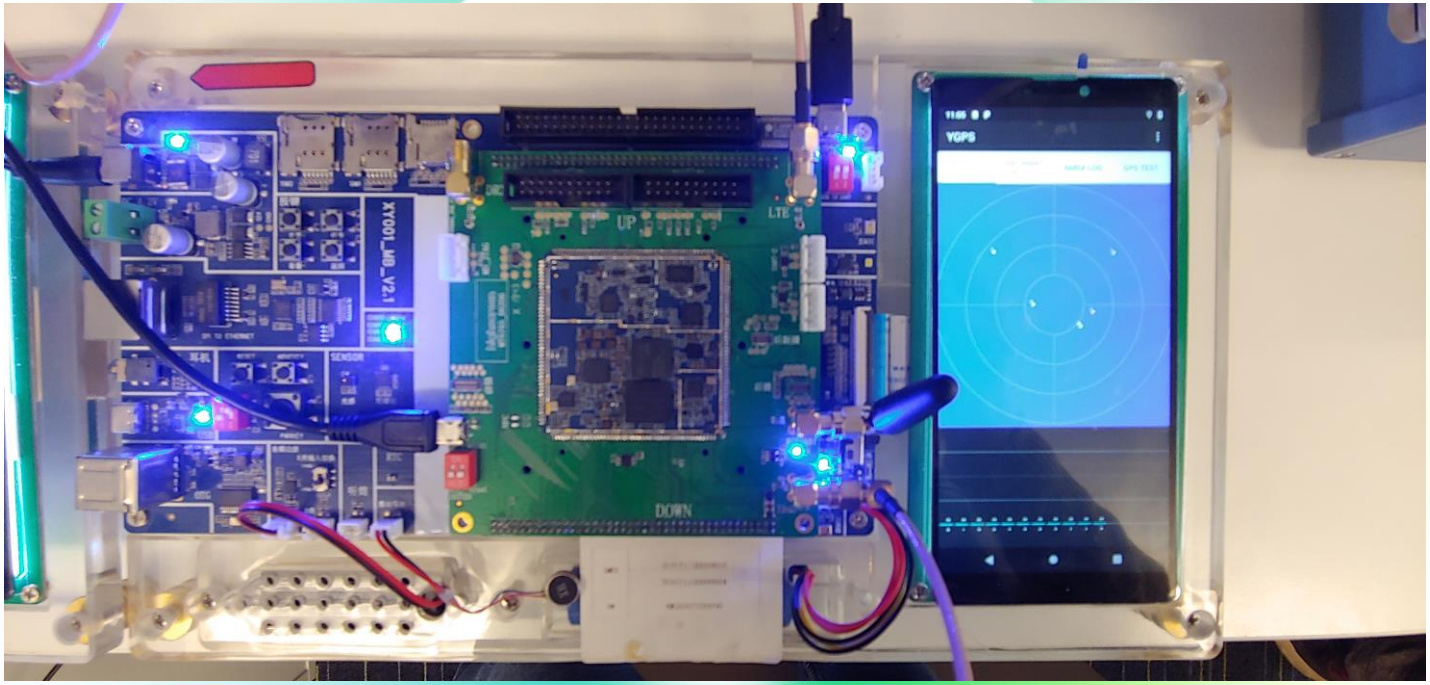


Тестирование сотовой сети



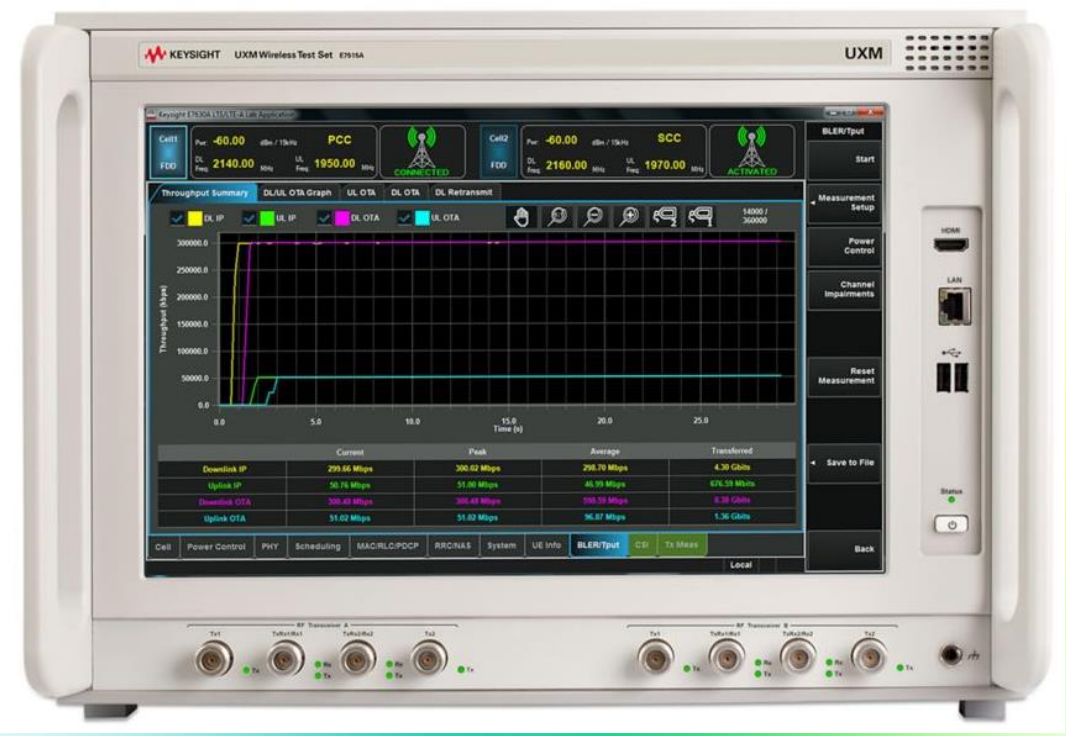


Телефон изнутри



kaspersky

Эмулятор базовой станции



kaspersky

Преимущества

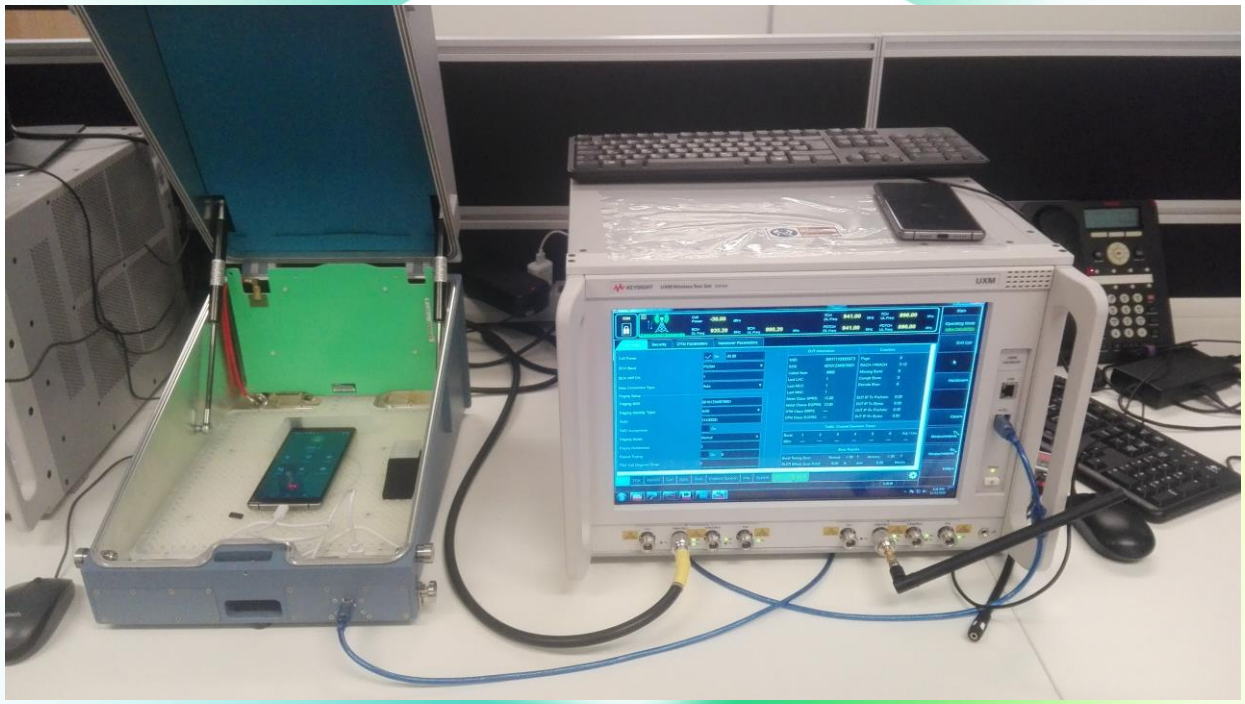
- Комплексное тестирование на телефоне
- Исключение влияния внешней среды
- Детальное исследования определённых режимов работы
- Повторяемость тестов

Интеграция в CI

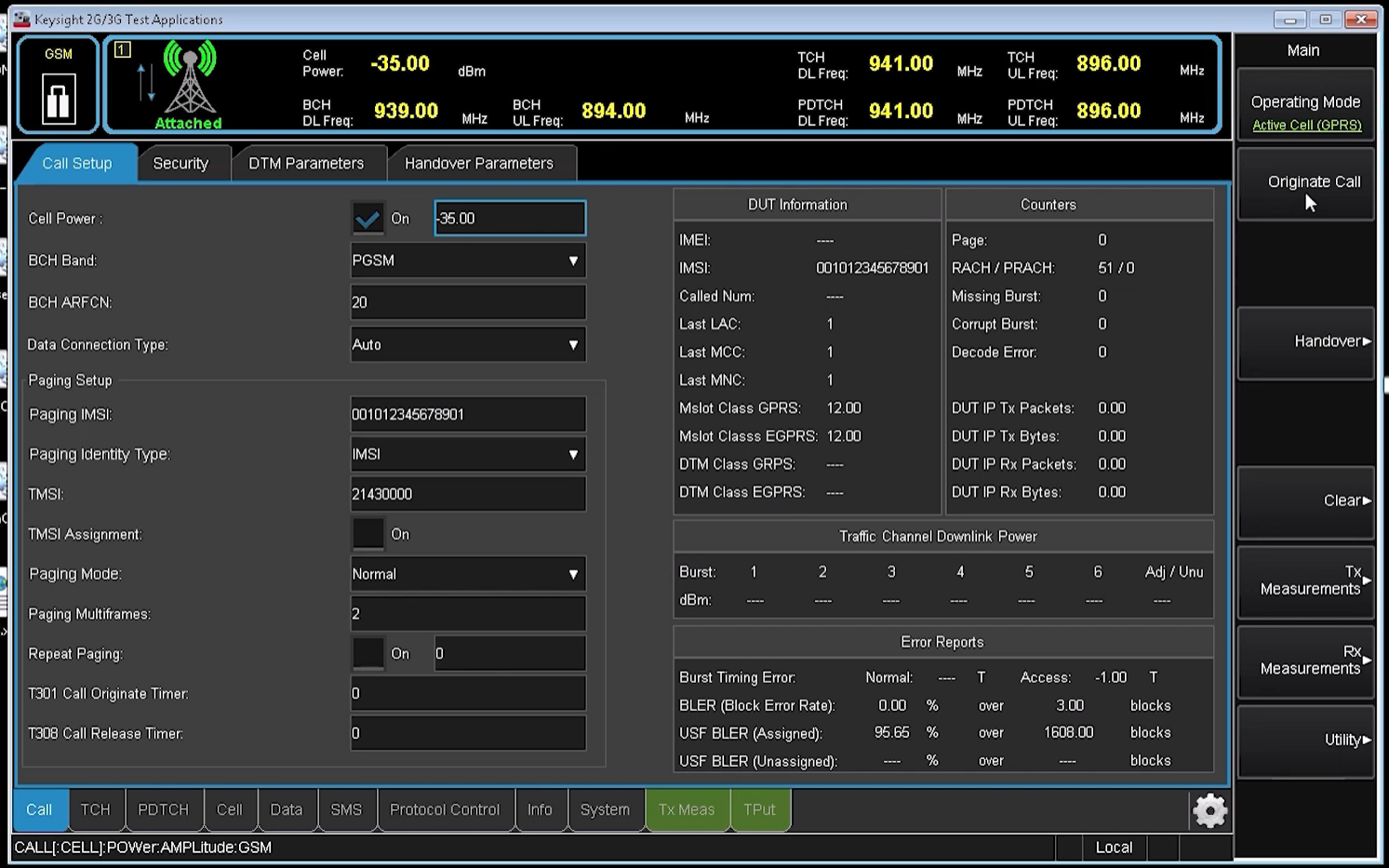
- Минимизация ручного труда
- Автоматизация управления прибором
- Интеграция в существующую систему

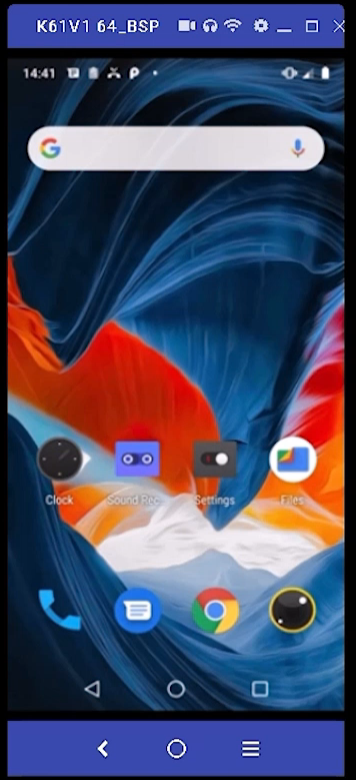
Схема стенда

Стенд

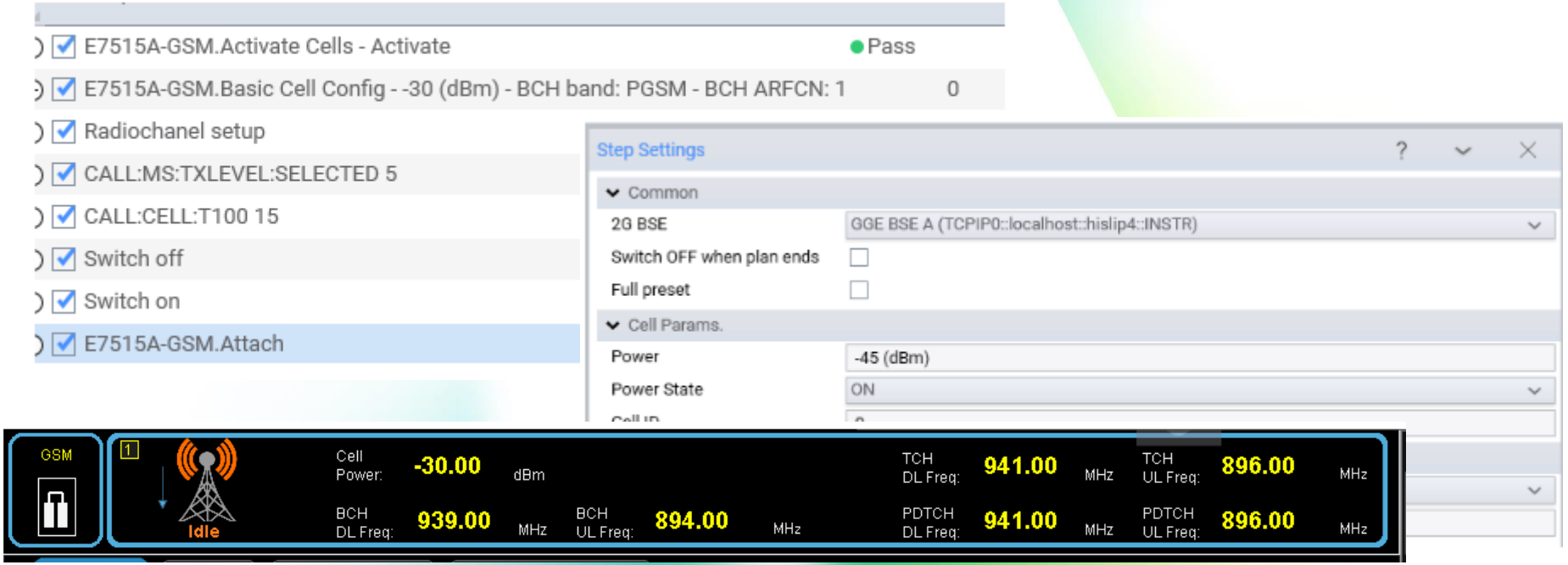


kaspersky


Keysight 2G/3G Test Applications
GSM
Attached
Cell Power: -35.00 dBm
BCH DL Freq: 939.00 MHz
BCH UL Freq: 894.00 MHz
TCH DL Freq: 941.00 MHz
TCH UL Freq: 896.00 MHz
PDTCH DL Freq: 941.00 MHz
PDTCH UL Freq: 896.00 MHz
Main
Operating Mode
Active Cell (GPRS)
Originate Call
Handover
Clear
Tx Measurements
Rx Measurements
Utility
Call Setup
Security
DTM Parameters
Handover Parameters
Cell Power : On -35.00
BCH Band: PGSM
BCH ARFCN: 20
Data Connection Type: Auto
Paging Setup
Paging IMSI: 001012345678901
Paging Identity Type: IMSI
TMSI: 21430000
TMSI Assignment: On
Paging Mode: Normal
Paging Multiframes: 2
Repeat Paging: On 0
T301 Call Originate Timer: 0
T308 Call Release Timer: 0
DUT Information
IMEI: ----
IMSI: 001012345678901
Called Num: ----
Last LAC: 1
Last MCC: 1
Last MNC: 1
Mslot Class GPRS: 12.00
Mslot Classs EGPRS: 12.00
DTM Class GRPS: ----
DTM Class EGPRS: ----
Counters
Page: 0
RACH / PRACH: 51 / 0
Missing Burst: 0
Corrupt Burst: 0
Decode Error: 0
DUT IP Tx Packets: 0.00
DUT IP Tx Bytes: 0.00
DUT IP Rx Packets: 0.00
DUT IP Rx Bytes: 0.00
Traffic Channel Downlink Power
Burst: 1 2 3 4 5 6 Adj / Unu
dBm: ---- ---- ---- ---- ---- ---- ----
Error Reports
Burst Timing Error: Normal: ---- T Access: -1.00 T
BLER (Block Error Rate): 0.00 % over 3.00 blocks
USF BLER (Assigned): 95.65 % over 1608.00 blocks
USF BLER (Unassigned): ---- % over ---- blocks
Call
TCH
PDTCH
Cell
Data
SMS
Protocol Control
Info
System
Tx Meas
TPut
CALL[:CELL]:POWer:AMPLitude:GSM
Local


K61V1 64_BSP
14:41

TAP (Test Automation Platform)



Что следует улучшить?

- Программное управление телефоном
- Запуск последовательности тестов

Android Debug Bridge (adb)

- Основные команды для тестов
- Унификация для нескольких телефонов
- Логирование

ADB – Звонки

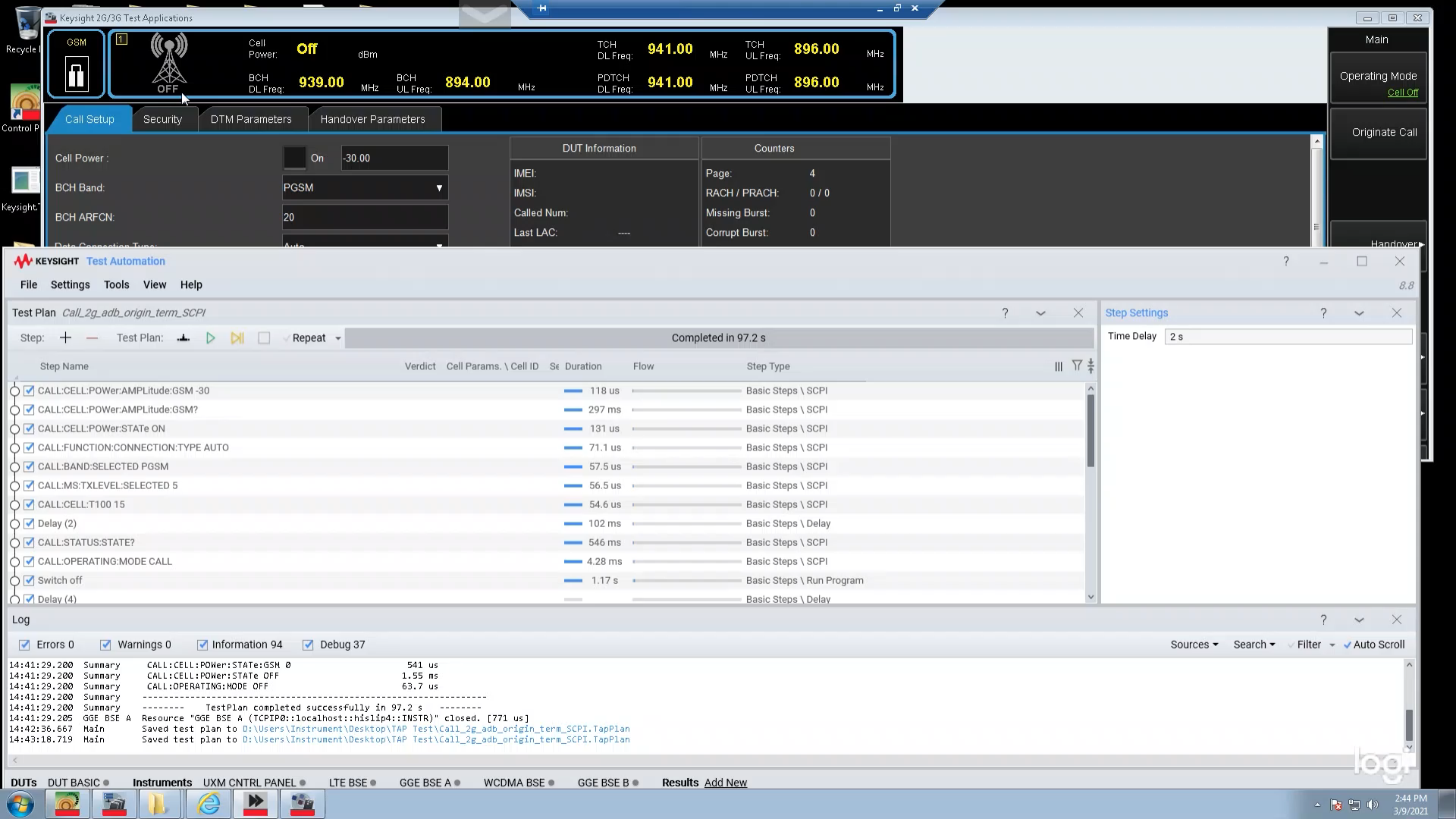
```
adb shell settings put global airplane_mode_on 1
adb shell am start -a android.intent.action.CALL -d
tel:+79687
adb shell input keyevent 6
adb shell am start -a
android.intent.action.CALL_PRIVILEGED -d tel:112
```

ADB – сообщения

```
adb shell settings put global airplane_mode_on 1
adb shell am broadcast -a
adb shell service call isms 7 i32 3 s16
"com.android.mms" s16 "+79001112233" s16 "null" s16
"'Hello Team KOS Mobile'" i32 0 i32 0
```

ADB – логи, передача данных

```
adb logcat -b radio
adb shell svc data enable
adb shell am start -a android.intent.action.CALL -d
tel:*102%23
```


Keysight 2G/3G Test Applications
GSM
Cell Power: Off dBm
BCH DL Freq: 939.00 MHz
BCH UL Freq: 894.00 MHz
TCH DL Freq: 941.00 MHz
TCH UL Freq: 896.00 MHz
PDTCH DL Freq: 941.00 MHz
PDTCH UL Freq: 896.00 MHz
OFF
Main
Operating Mode
Cell Off
Originate Call
Handover
Call Setup
Security
DTM Parameters
Handover Parameters
Cell Power :
On
-30.00
BCH Band:
PGSM
BCH ARFCN:
20
DUT Information
IMEI:
IMSI:
Called Num:
Last LAC:
Counters
Page: 4
RACH / PRACH: 0 / 0
Missing Burst: 0
Corrupt Burst: 0
KEYSIGHT Test Automation
File Settings Tools View Help
8.8
Test Plan Call_2g_adb_origin_term_SCPI
Step: Test Plan: Repeat
Completed in 97.2 s
Step Settings
Time Delay 2 s
Step Name Verdict Cell Params. \ Cell ID Se Duration Flow Step Type
CALL:CELL:POWer:AMPLitude:GSM -30 118 us Basic Steps \ SCPI
CALL:CELL:POWer:AMPLitude:GSM? 297 ms Basic Steps \ SCPI
CALL:CELL:POWer:STATe ON 131 us Basic Steps \ SCPI
CALL:FUNCTION:CONNECTION:TYPE AUTO 71.1 us Basic Steps \ SCPI
CALL:BAND:SELECTED PGSM 57.5 us Basic Steps \ SCPI
CALL:MS:TXLEVEL:SELECTED 5 56.5 us Basic Steps \ SCPI
CALL:CELL:T100 15 54.6 us Basic Steps \ SCPI
Delay (2) 102 ms Basic Steps \ Delay
CALL:STATUS:STATE? 546 ms Basic Steps \ SCPI
CALL:OPERATING:MODE CALL 4.28 ms Basic Steps \ SCPI
Switch off 1.17 s Basic Steps \ Run Program
Delay (4) Basic Steps \ Delay
Log
Errors 0 Warnings 0 Information 94 Debug 37
Sources Search Filter Auto Scroll
14:41:29.200 Summary CALL:CELL:POWer:STATe:GSM 0 541 us
14:41:29.200 Summary CALL:CELL:POWer:STATe OFF 1.55 ms
14:41:29.200 Summary CALL:OPERATING:MODE OFF 63.7 us
14:41:29.200 Summary ----------
14:41:29.200 Summary -------- TestPlan completed successfully in 97.2 s --------
14:41:29.205 GGE BSE A Resource "GGE BSE A (TCPIP0::localhost::hislip4::INSTR)" closed. [771 us]
14:42:36.667 Main Saved test plan to D:\Users\Instrument\Desktop\TAP Test\Call_2g_adb_origin_term_SCPI.TapPlan
14:43:18.719 Main Saved test plan to D:\Users\Instrument\Desktop\TAP Test\Call_2g_adb_origin_term_SCPI.TapPlan
DUTs DUT BASIC
Instruments UXM CNTRL PANEL LTE BSE GGE BSE A WCDMA BSE GGE BSE B
Results Add New
2:44 PM
3/9/2021

Интеграция в CI

Преимущества

+ Удобный интерфейс
+ Легко отлаживать
+ Можно запускать из командной строки
+ Конфигурирование прибора

Интеграция в CI

Недостатки

- Формируется очередь
- Долго выполняется тест
- Много ручного труда
- Тесты должны храниться непосредственно на приборе
- Нет управления из Python

OpenTAP – python plugin

```
$TAP_PATH/Packages/Python/
```

Интеграция в CI

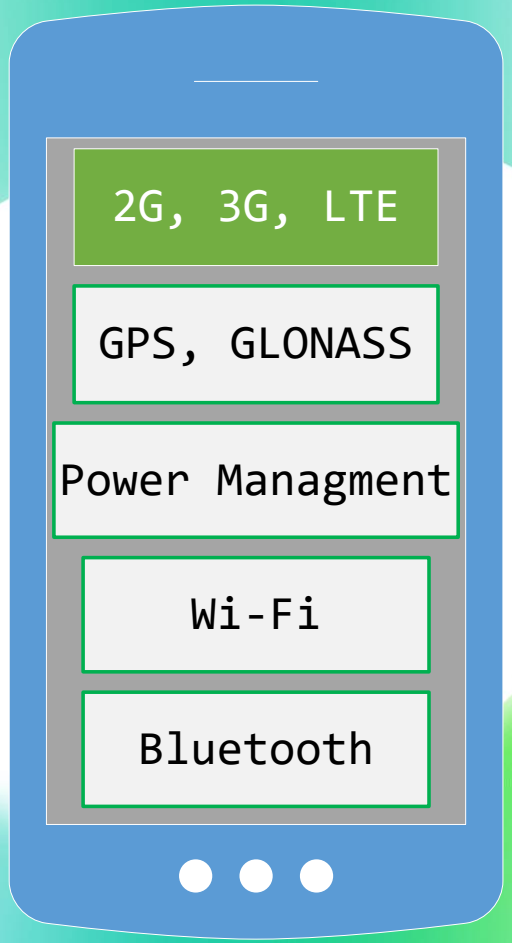
Преимущества

+ Программное управление телефоном
+ Конфигурирование прибора
+ Запуск из Python

Интеграция в CI

Недостатки

- Управление через прибор
- Ограничения конфигурирования оборудования
- Схема не масштабируется


2G, 3G, LTE
GPS, GLONASS
Power Managment
Wi-Fi
Bluetooth

VISA + HiSLIP

VISA – Virtual Instrument Software Architecture

HiSLIP – High-Speed LAN Instrument Protocol

pyvisa

```
import pyvisa
def __init__(self, address: str, visa_lib:
str):
    self.rm = pyvisa.ResourceManager(visa_lib)
    self.address = address
    self.ControlPanel: ControlPanel = None
    self.TransceiverA: Transceiver = None
    self.TransceiverB: Transceiver = None
```

Подключение к панели управления

```
def control_panel_start(self):
    self.ControlPanel = ControlPanel(self.rm,
    'TCPIP0::' + self.address +
    '::hislip0::INSTR’)
    logging.info(self.ControlPanel, "Control
    Panel hasn't been connected")
```

Настройка трансивера

```
def configuration_set_3g(self, power_cell: int = -50,
                                 cable_correction: int = 0):
    self.TransceiverA = Transceiver3G(rm=self.rm,
    port='TCPIP0::' + self.address + '::hislip4::INSTR’)
    self.TransceiverA.configuration_set(power_cell=power_c
    ell, cable_correction=cable_correction)
```

Что-то знакомое

```
div > div:nth-of-type(2) div > div:nth-child(2)

#main-panel > div.slideshow-info-container >
div.notranslate.transcript.add-padding-right.j-transcript
> ol > li:nth-child(1)
```

SCPI - Standard Commands for Programmable Instruments

```
SOURce:POWer[:LEVel]:SLOPe[:DATA]
SENSe:FREQuency:CENTer
SENSe:FREQuency:STARt?
SENS:FREQ:STAR 1 MHZ;STOP 2MHZ
```

- Формат команд
- Условные обозначения
- Запрос настроек параметров
- Терминаторы команд SCPI

SCPI параметризация команд

```
SENSe:FREQuency:STARt 1000000
SENSe:FREQuency:STARt 1 MHz
DISPlay:ENABle OFF
MMEMory:STORe "state01.sta"
CALC:LIMit:DATA
2,1,1E9,3E9,0,0,2,1E9,3E9,-3,-3
```

- Числовые параметры
- Дискретные параметры
- Строк ASCII
- Числовые списки

kaspersky

Пример использования SCPI команд для отправки SMS

```
def __call_sms_ptp_text_cust(self, text: str):
    ans =
    self.instrument.write('CALL:SMService:PTPoint:
    TEXT:CUSTom "%s"' % text)
```

Пример использования SCPI команд для вызовов

```
def __call_orig(self):
    ans = self.instrument.write('CALL:ORIGinate’)

def __call_end(self):
    ans = self.instrument.write('CALL:END')
```

Интеграция в CI

Преимущества

+ Управление по сети
+ Распараллеливание работ
+ Подключение к CI

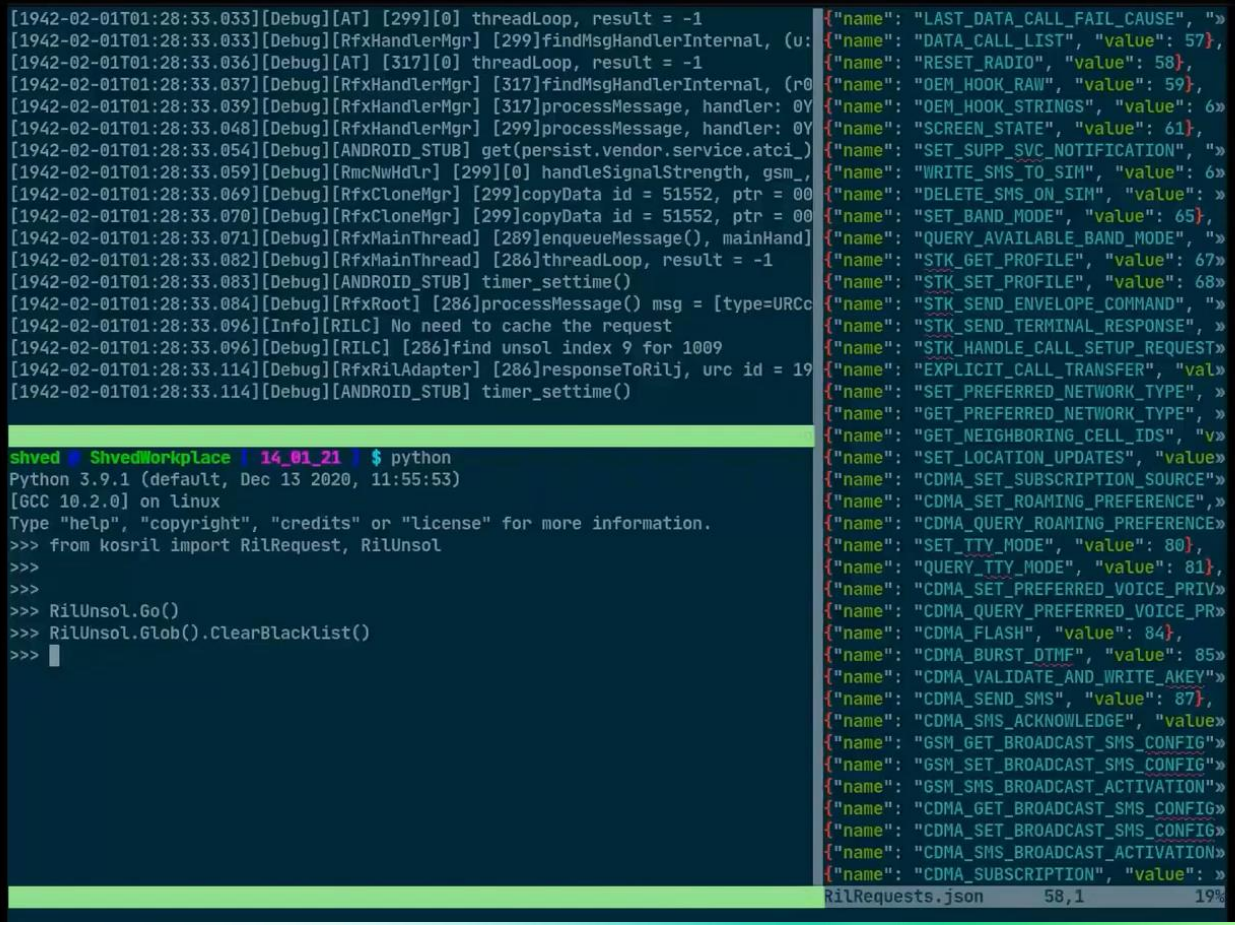
Интеграция в CI

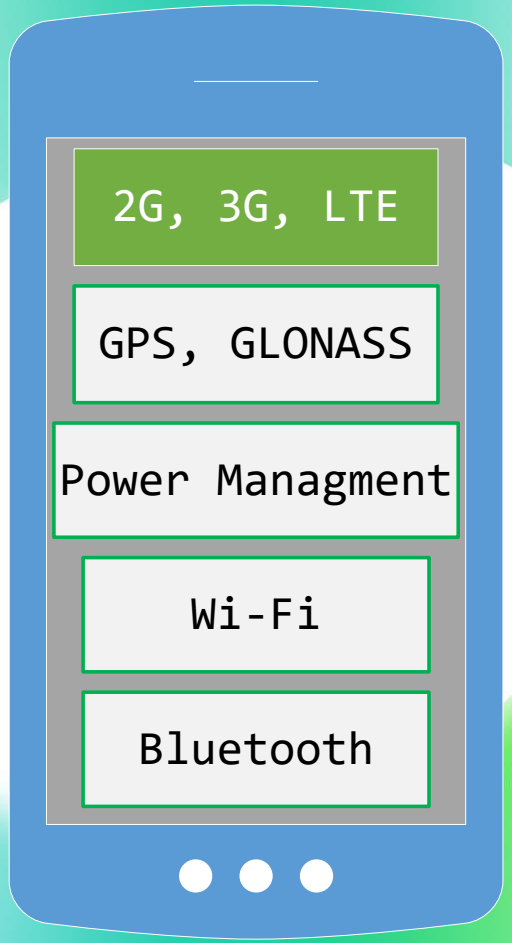
Недостатки

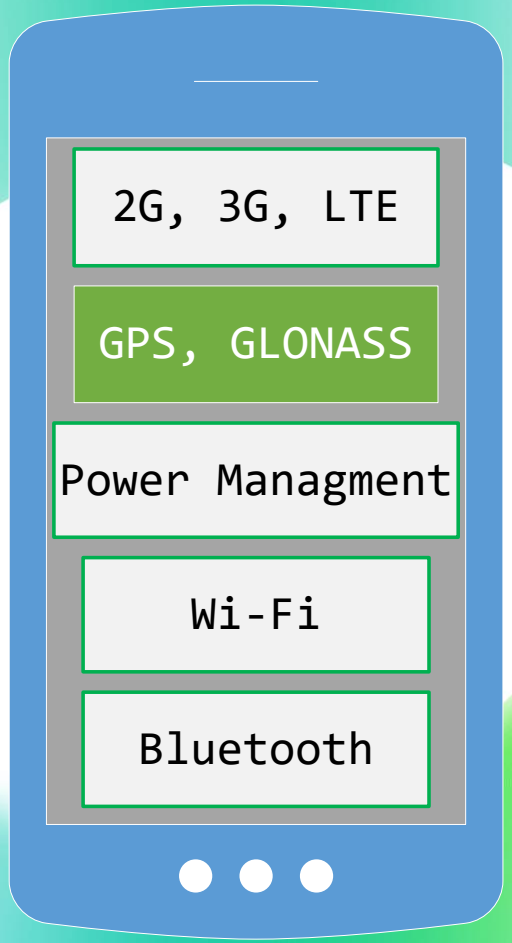
- Ручная отладка
- Высокий порог вхождения в тесты

Путь к автоматизации

- Построение стендов
- Ручное тестирование
- Проверка тестов на телефоне под управлением Android
- Автоматизация тестов через TAP
- Переход к OpenTAP Python
- Удалённое управление прибором при помощи SCPI команд, HiSLIP + VISA протоколов


[1942-02-01T01:28:33.033][Debug][AT] [299][0] threadLoop, result = -1
[1942-02-01T01:28:33.033][Debug][RfxHandlerMgr] [299]findMsgHandlerInternal, (v:
[1942-02-01T01:28:33.036][Debug][AT] [317][0] threadLoop, result = -1
[1942-02-01T01:28:33.037][Debug][RfxHandlerMgr] [317]findMsgHandlerInternal, (r0
[1942-02-01T01:28:33.039][Debug][RfxHandlerMgr] [317]processMessage, handler: 0Y
[1942-02-01T01:28:33.048][Debug][RfxHandlerMgr] [299]processMessage, handler: 0Y
[1942-02-01T01:28:33.054][Debug][ANDROID_STUB] get(persist.vendor.service.atci_)
[1942-02-01T01:28:33.059][Debug][RmcNwHdlr] [299][0] handleSignalStrength, gsm_,
[1942-02-01T01:28:33.069][Debug][RfxCloneMgr] [299]copyData id = 51552, ptr = 00
[1942-02-01T01:28:33.070][Debug][RfxCloneMgr] [299]copyData id = 51552, ptr = 00
[1942-02-01T01:28:33.071][Debug][RfxMainThread] [289]enqueueMessage(), mainHand]
[1942-02-01T01:28:33.082][Debug][RfxMainThread] [286]threadLoop, result = -1
[1942-02-01T01:28:33.083][Debug][ANDROID_STUB] timer_settime()
[1942-02-01T01:28:33.084][Debug][RfxRoot] [286]processMessage() msg = [type=URCc
[1942-02-01T01:28:33.096][Info][RILC] No need to cache the request
[1942-02-01T01:28:33.096][Debug][RILC] [286]find unsol index 9 for 1009
[1942-02-01T01:28:33.114][Debug][RfxRilAdapter] [286]responseToRilj, urc id = 19
[1942-02-01T01:28:33.114][Debug][ANDROID_STUB] timer_settime()
shved @ ShvedWorkplace] 14_01_21] $ python
Python 3.9.1 (default, Dec 13 2020, 11:55:53)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> from kosril import RilRequest, RilUnsol
>>>
>>>
>>> RilUnsol.Go()
>>> RilUnsol.Glob().ClearBlacklist()
>>>
{"name": "LAST_DATA_CALL_FAIL_CAUSE", "»
{"name": "DATA_CALL_LIST", "value": 57},
{"name": "RESET_RADIO", "value": 58},
{"name": "OEM_HOOK_RAW", "value": 59},
{"name": "OEM_HOOK_STRINGS", "value": 6»
{"name": "SCREEN_STATE", "value": 61},
{"name": "SET_SUPP_SVC_NOTIFICATION", "»
{"name": "WRITE_SMS_TO_SIM", "value": 6»
{"name": "DELETE_SMS_ON_SIM", "value": »
{"name": "SET_BAND_MODE", "value": 65},
{"name": "QUERY_AVAILABLE_BAND_MODE", "»
{"name": "STK_GET_PROFILE", "value": 67»
{"name": "STK_SET_PROFILE", "value": 68»
{"name": "STK_SEND_ENVELOPE_COMMAND", "»
{"name": "STK_SEND_TERMINAL_RESPONSE", »
{"name": "STK_HANDLE_CALL_SETUP_REQUEST»
{"name": "EXPLICIT_CALL_TRANSFER", "val»
{"name": "SET_PREFERRED_NETWORK_TYPE", »
{"name": "GET_PREFERRED_NETWORK_TYPE", »
{"name": "GET_NEIGHBORING_CELL_IDS", "v»
{"name": "SET_LOCATION_UPDATES", "value»
{"name": "CDMA_SET_SUBSCRIPTION_SOURCE"»
{"name": "CDMA_SET_ROAMING_PREFERENCE",»
{"name": "CDMA_QUERY_ROAMING_PREFERENCE»
{"name": "SET_TTY_MODE", "value": 80},
{"name": "QUERY_TTY_MODE", "value": 81},
{"name": "CDMA_SET_PREFERRED_VOICE_PRIV»
{"name": "CDMA_QUERY_PREFERRED_VOICE_PR»
{"name": "CDMA_FLASH", "value": 84},
{"name": "CDMA_BURST_DTMF", "value": 85»
{"name": "CDMA_VALIDATE_AND_WRITE_AKEY"»
{"name": "CDMA_SEND_SMS", "value": 87},
{"name": "CDMA_SMS_ACKNOWLEDGE", "value»
{"name": "GSM_GET_BROADCAST_SMS_CONFIG"»
{"name": "GSM_SET_BROADCAST_SMS_CONFIG"»
{"name": "GSM_SMS_BROADCAST_ACTIVATION"»
{"name": "CDMA_GET_BROADCAST_SMS_CONFIG»
{"name": "CDMA_SET_BROADCAST_SMS_CONFIG»
{"name": "CDMA_SMS_BROADCAST_ACTIVATION»
{"name": "CDMA_SUBSCRIPTION", "value": »
RilRequests.json 58,1 19%


2G, 3G, LTE
GPS, GLONASS
Power Managment
Wi-Fi
Bluetooth


2G, 3G, LTE
GPS, GLONASS
Power Managment
Wi-Fi
Bluetooth

Что тестируем

- Точность определения координат
- Время обнаружения спутников
- Различные конфликтные ситуации

GPS эмулятор

Тестирование работы
с геопозицией.

GPS эмулятор

```
def set_location(self, lat: float, lon: float, hgt: float):

        self.process = subprocess.Popen([self.path + 'GPS_emulator’,
                                         '-e', self.path + self.brdc,
                                         '-l', ','.join([str(lat),
                                         str(lon), str(hgt)])],
                                         stdout=subprocess.PIPE)

        output = self.process.stdout.readline().decode("UTF-8")
        while "GPS signal generator is ready" not in output:
                output = self.process.stdout.readline().decode("UTF-8")
                info(output)
```

kaspersky

GPS эмулятор

```
Tests passed: 0 of 1 test
DEBUG    root:__init__.py:50 Broadcasting: Intent { act=android.intent.action.AIRPLANE_MODE flg=0x400000 (has extras) }
                             Security exception: Permission Denial: not allowed to send broadcast android.intent.action.AIRPLANE_MODE from pid=13257, uid=2000

                             java.lang.SecurityException: Permission Denial: not allowed to send broadcast android.intent.action.AIRPLANE_MODE from pid=13257, uid=2000
                                at com.android.server.am.ActivityManagerService.broadcastIntentLocked(ActivityManagerService.java:21537)
                                at com.android.server.am.ActivityManagerService.broadcastIntent(ActivityManagerService.java:22174)
                                at com.android.server.am.ActivityManagerShellCommand.runSendBroadcast(ActivityManagerShellCommand.java:690)
                                at com.android.server.am.ActivityManagerShellCommand.onCommand(ActivityManagerShellCommand.java:174)
                                at android.os.ShellCommand.exec(ShellCommand.java:103)
                                at com.android.server.am.ActivityManagerService.onShellCommand(ActivityManagerService.java:16216)
                                at android.os.Binder.shellCommand(Binder.java:634)
                                at android.os.Binder.onTransact(Binder.java:532)
                                at android.app.IActivityManager$Stub.onTransact(IActivityManager.java:3569)
                                at com.android.server.am.ActivityManagerService.onTransact(ActivityManagerService.java:3350)
                                at android.os.Binder.execTransact(Binder.java:731)
DEBUG    root:__init__.py:57
DEBUG    root:__init__.py:59 Broadcasting: Intent { act=android.intent.action.AIRPLANE_MODE flg=0x400000 (has extras) }
                             Security exception: Permission Denial: not allowed to send broadcast android.intent.action.AIRPLANE_MODE from pid=13282, uid=2000

                             java.lang.SecurityException: Permission Denial: not allowed to send broadcast android.intent.action.AIRPLANE_MODE from pid=13282, uid=2000
                                at com.android.server.am.ActivityManagerService.broadcastIntentLocked(ActivityManagerService.java:21537)
                                at com.android.server.am.ActivityManagerService.broadcastIntent(ActivityManagerService.java:22174)
                                at com.android.server.am.ActivityManagerShellCommand.runSendBroadcast(ActivityManagerShellCommand.java:690)
                                at com.android.server.am.ActivityManagerShellCommand.onCommand(ActivityManagerShellCommand.java:174)
                                at android.os.ShellCommand.exec(ShellCommand.java:103)
                                at com.android.server.am.ActivityManagerService.onShellCommand(ActivityManagerService.java:16216)
                                at android.os.Binder.shellCommand(Binder.java:634)
                                at android.os.Binder.onTransact(Binder.java:532)
                                at android.app.IActivityManager$Stub.onTransact(IActivityManager.java:3569)
                                at com.android.server.am.ActivityManagerService.onTransact(ActivityManagerService.java:3350)
                                at android.os.Binder.execTransact(Binder.java:731)
INFO     root:__init__.py:46 Opening and initializing device...
INFO     root:__init__.py:49 deviceName: LimeSDR-USB
```

PyCharm 2020.2.5 available
Update...

kaspersky

Интеграция в CI

Преимущества

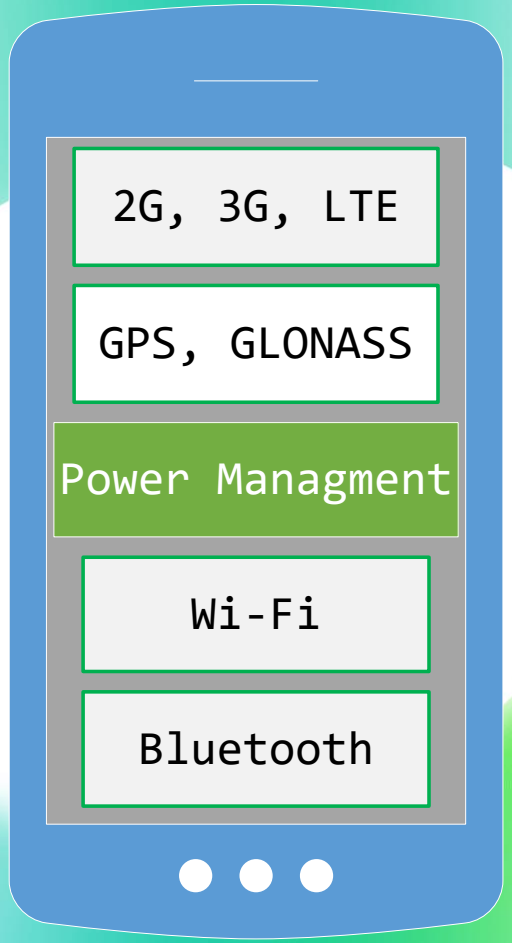
+ Автоматизированное управление GPS эмулятором
+ Открытый исходный код

kaspersky

Интеграция в CI

Недостатки

- Новый протокол взаимодействия
- Низкая точность геопозиционирования
- Высокие требования к ПК


2G, 3G, LTE
GPS, GLONASS
Power Managment
Wi-Fi
Bluetooth

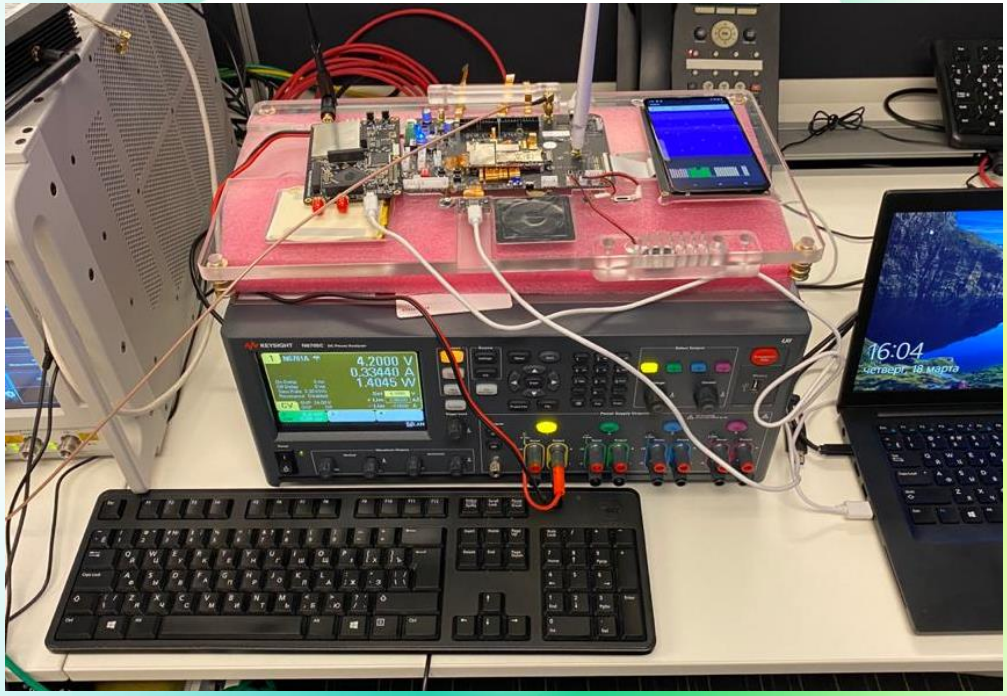
Что тестируем

- Потребление узлов
- Работу при граничных уровнях заряда
- Работа зарядки
- Правильная работа ОС с аккумулятором

Power Monitor

kaspersky

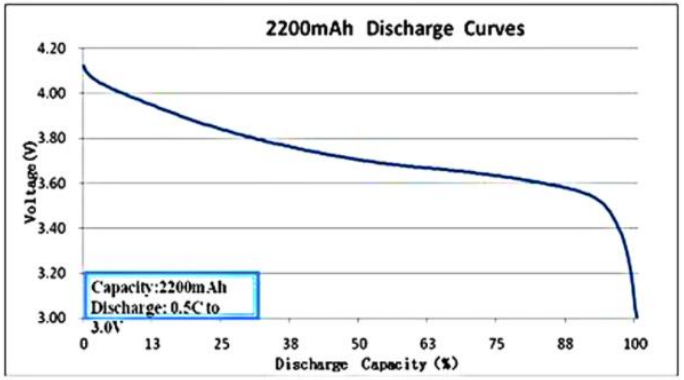
Power Monitor



kaspersky

Power Monitor

- Оценка энергопотребления узлов
- Имитация батареи
- Имитация зарядного устройства
- Повторяемость
- Интеграция в CI



Интеграция в CI

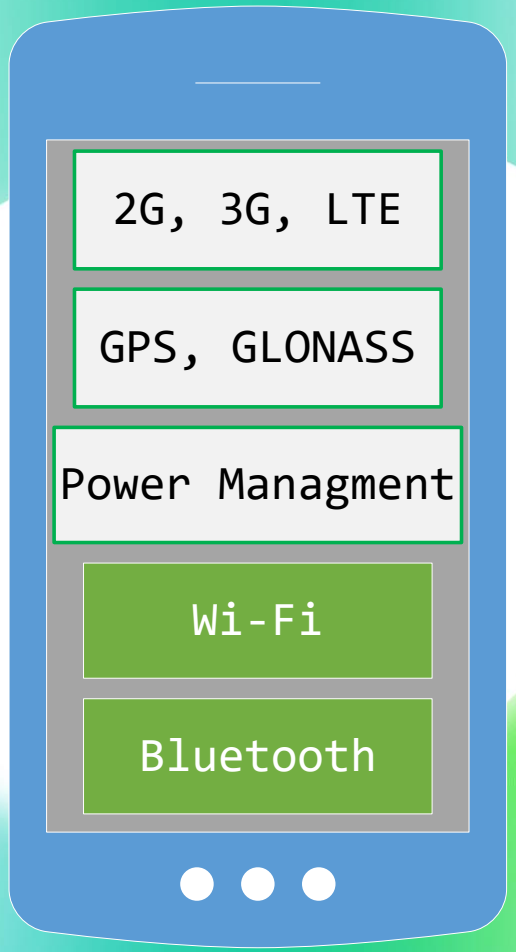
Преимущества

+ Автоматизированное управление по сети
+ Высокая повторяемость тестов
+ Независимость от износа телефона
+ Моделирование нестабильностей и сложных сценариев

Интеграция в CI

Недостатки

- Требуются отладочные платы
- Высокая стоимость


2G, 3G, LTE
GPS, GLONASS
Power Managment
Wi-Fi
Bluetooth

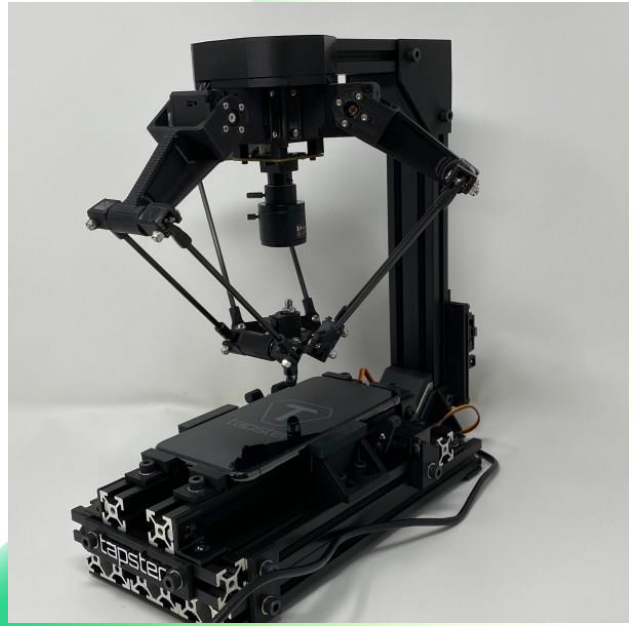
Bluetooth, WiFi emulator

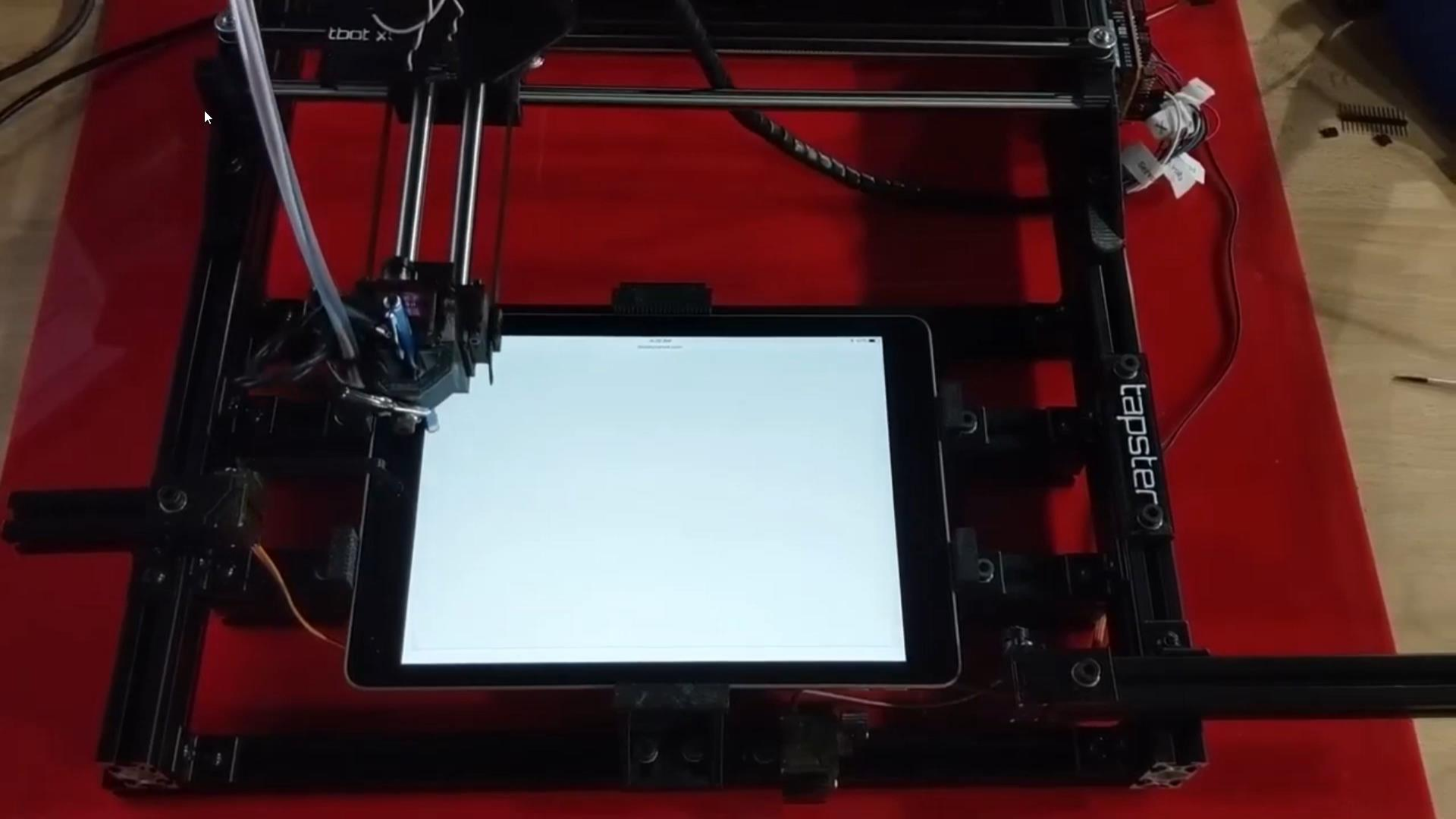
- Контроль качества работы драйверов
- Исключение влияния окружающей среды
- Проверка граничных значений
- Моделирование нестандартных ситуаций

Bluetooth и Wi-Fi

Сквозные тесты

- Все пользуются телефоном по-разному
- Добавим в CI сложные сценарии




tbot x
tapstar

Выводы

- Эмуляторы используются для CI
- Генераторы управляются по сети
- Схема стендов расширяемая
- Высокая повторяемость тестов
- Сложные сценарии тестирования
- Проверки с высокой точностью
- Регресс автоматизирован

Рекомендации

- Начинать с эмуляторов
- Проводить глубокий анализ тестовых сценариев
- Применять общий подход к оборудованию (VISA + HiSLIP)
- Строить CI одинаково для различных проектов
- Следить за разумностью автоматизации

Ссылки

https://os.kaspersky.ru/
https://www.qemu.org/
https://wiki.qemu.org/Documentation/QMP
https://pypi.org/project/qmp/
https://github.com/utepnetlab/opentap
https://en.wikipedia.org/wiki/Standard_Commands_for_Programmable_Instruments
https://habr.com/ru/article/499746/#rec186469108

Q&A

Questions and Answers

Надежда Давыдова
Nadezhda.Davydova@kaspersky.com