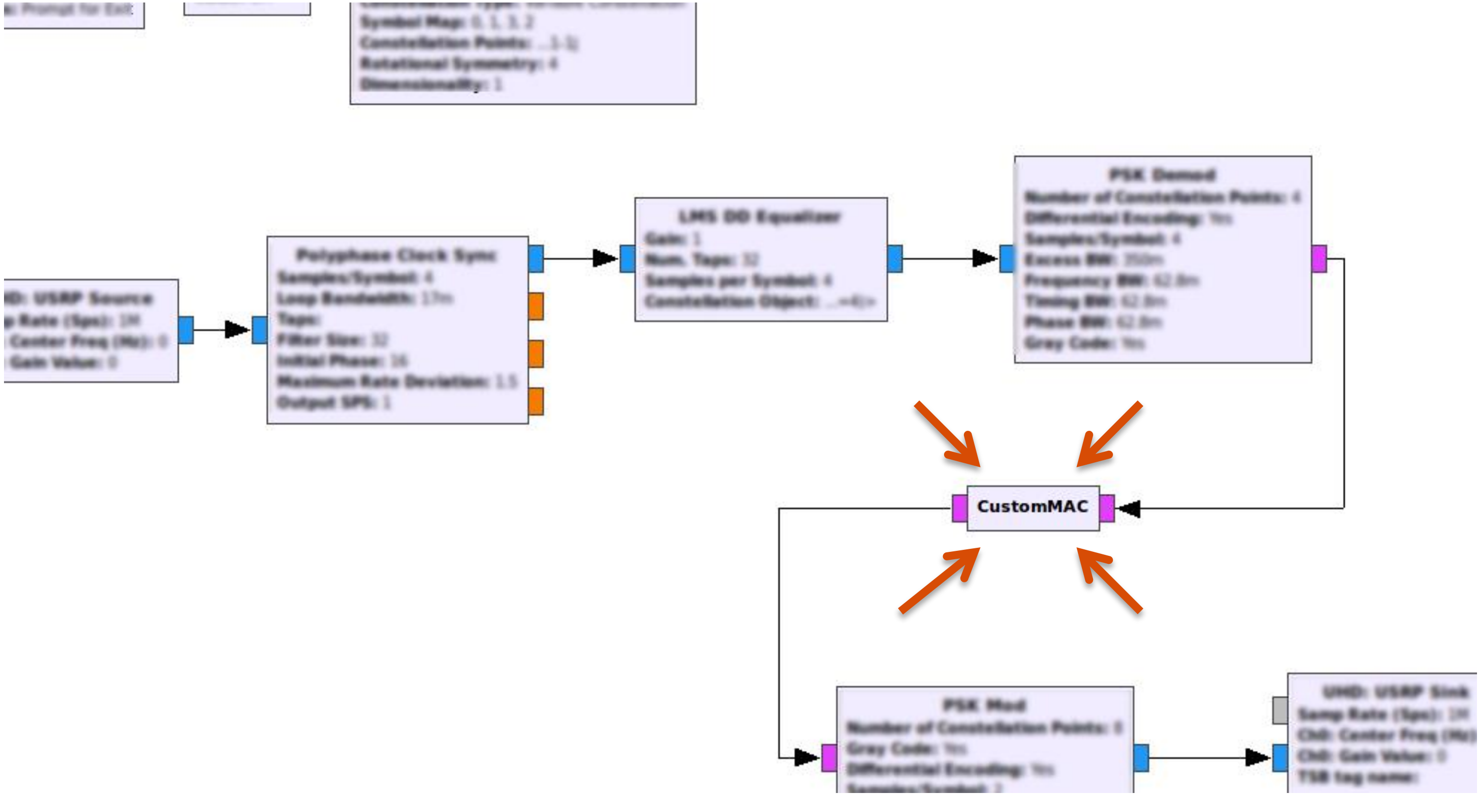
MAC Processor Development for GNURadio

Jarriel Cook

The Johns Hopkins Applied Physics Lab

APL
JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

Motivation



: Prompt for Exit

Symbol Map: 0, 1, 3, 2
Constellation Points: ...1-1j]
Rotational Symmetry: 4
Dimensionality: 1

D: USRP Source
p Rate (Sps): 1M
Center Freq (Hz): 0
Gain Value: 0

Polyphase Clock Sync
Samples/Symbol: 4
Loop Bandwidth: 17m
Taps:
Filter Size: 32
Initial Phase: 16
Maximum Rate Deviation: 1.5
Output SPS: 1

LMS DD Equalizer
Gain: 1
Num. Taps: 32
Samples per Symbol: 4
Constellation Object: ...>4()>

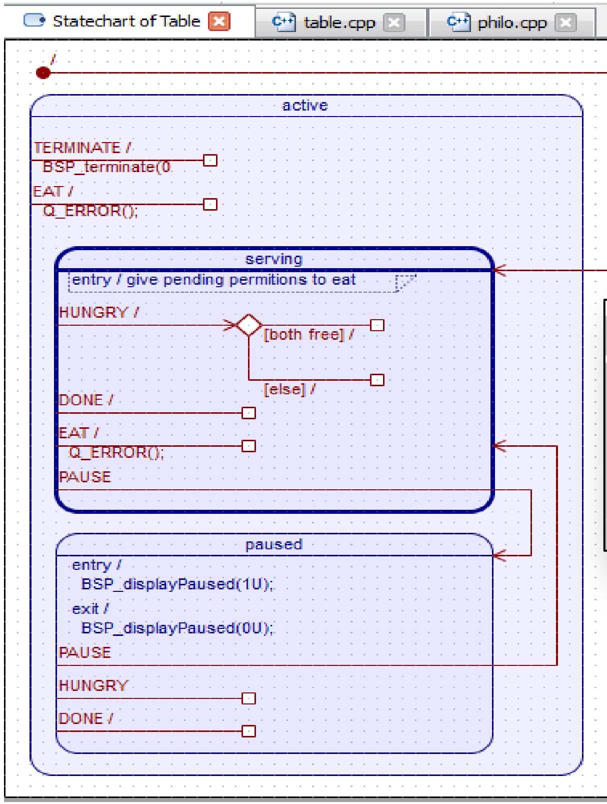
PSK Demod
Number of Constellation Points: 4
Differential Encoding: Yes
Samples/Symbol: 4
Excess BW: 350m
Frequency BW: 62.8m
Timing BW: 62.8m
Phase BW: 62.8m
Gray Code: Yes

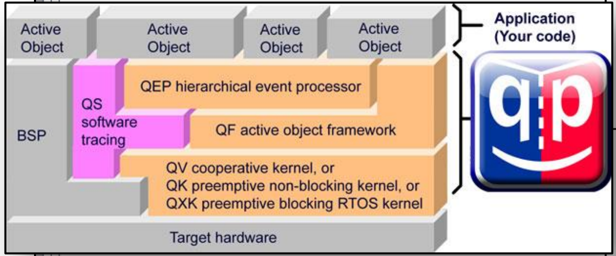
CustomMAC

PSK Mod
Number of Constellation Points: 8
Gray Code: Yes
Differential Encoding: Yes
Samples/Symbol: 2

UHD: USRP Sink
Samp Rate (Sps): 1M
Ch0: Center Freq (Hz): 0
Ch0: Gain Value: 0
TSB tag name:

A Different Approach




Active Object
Active Object
Active Object
Active Object
Application (Your code)
BSP
QS software tracing
QEP hierarchical event processor
QF active object framework
QV cooperative kernel, or
QK preemptive non-blocking kernel, or
QXK preemptive blocking RTOS kernel
Target hardware
q:p

GENERAL DYNAMICS

Without using QP™, I don't believe we could have delivered on our given schedule dates with the same level of quality."

Jeff Karau, Sr. Software Engineer, General Dynamics C4 Systems

Honeywell

"Quantum Leaps software has revolutionized not just the way we write our software, but the way we approach our design. It is intuitive, easy to implement and comes in an ***incredibly small package.****"*

Chad Koster, Software Engineer, Honeywell

I used the traditional RTOS approach for about 10 years. With the real-time debug log of QP/Spy™ and the ability to see what is going on in the logic flow, the code is very easy to debug and modify. It makes the code very modular and deterministic... You end up talking about the codes logic flow, and not the semantics of the software. QP™ is a great product."

Paul Walker, Software Engineer, EIM Controls, Inc.

All company names and logos mentioned herein are the trademarks of their respective owners.

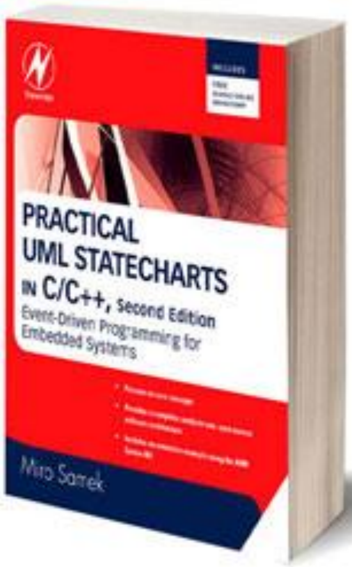
About QP

- **Quantum Leaps**
  - **Company that created the state machine software**
  - **Website: https://www.state-machine.com/**
- **QP Frameworks**
  - **A collection of ports that support building of event-driven applications**
  - **Support for C and C++, ports for Bare-Metal ARM, QT GUI, RTOS, etc.**
  - **Over 15 years of continuous development and 60,000 downloads a year**
  - **Licensed under GPLv3 or a commercial license**
- **QM**
  - **Quantum Modelling Tool**
  - **Model based design tool which auto-generates QEP source code**
- **QTools**
  - **QSpy is of primary interest for run-time tracing**

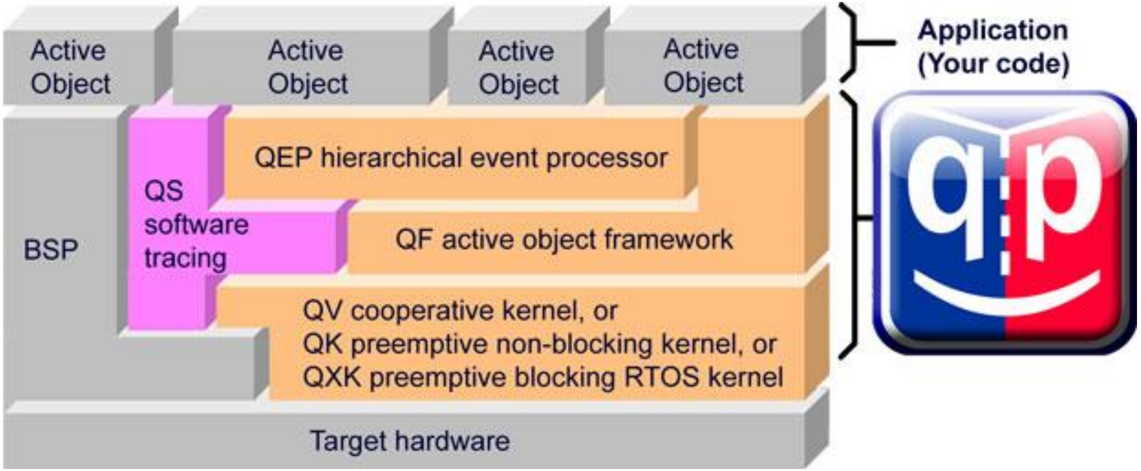
Provides a Framework For:

- **Thread-per-state-machine execution context**
- **Event queues**
- **Event based timing services**
- **Run-time execution tracing**
- ***Graphical* model-based design**

Documentation



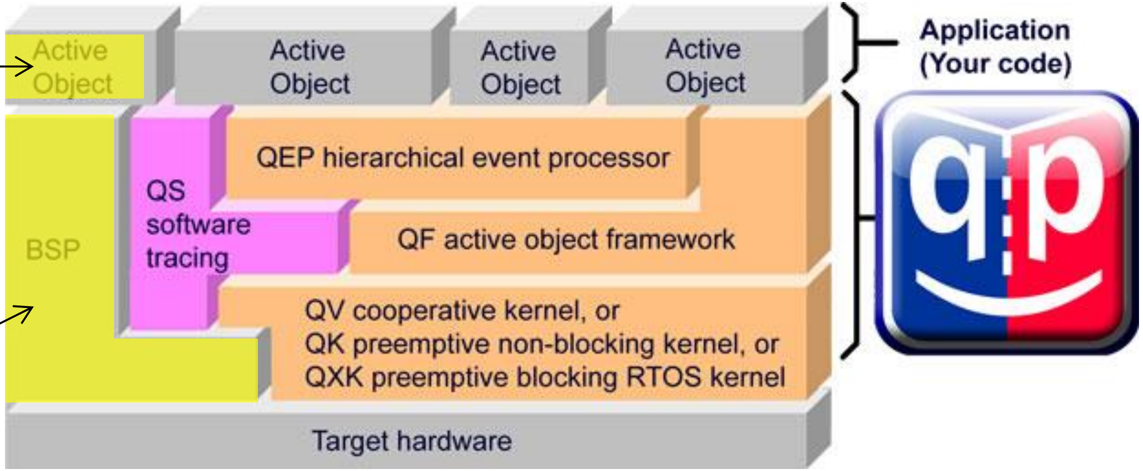
QP Architecture


Active Object
Active Object
Active Object
Active Object
Application (Your code)
BSP
QS software tracing
QEP hierarchical event processor
QF active object framework
QV cooperative kernel, or
QK preemptive non-blocking kernel, or
QXK preemptive blocking RTOS kernel
Target hardware
q:p

QP + CHEESY MAC

CHEESY MAC provides “extension” objects

CHEESY MAC provides the BSP for POSIX targets

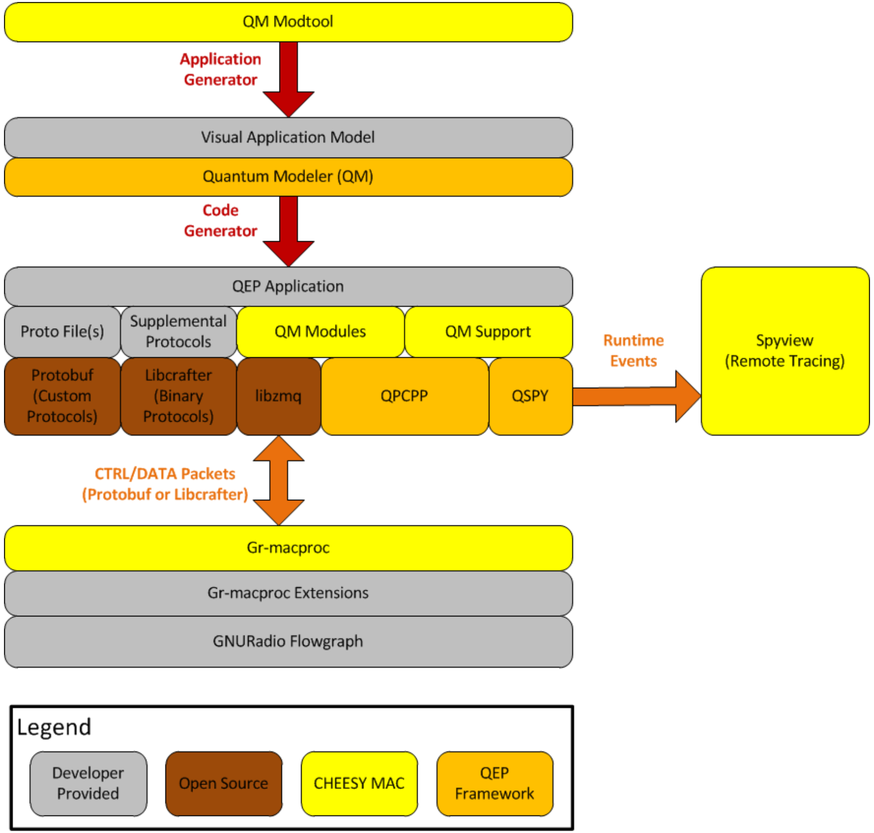

Active Object
Active Object
Active Object
Active Object
Application (Your code)
BSP
QS software tracing
QEP hierarchical event processor
QF active object framework
QV cooperative kernel, or
QK preemptive non-blocking kernel, or
QXK preemptive blocking RTOS kernel
Target hardware
q:p

What is CHEESY MAC?

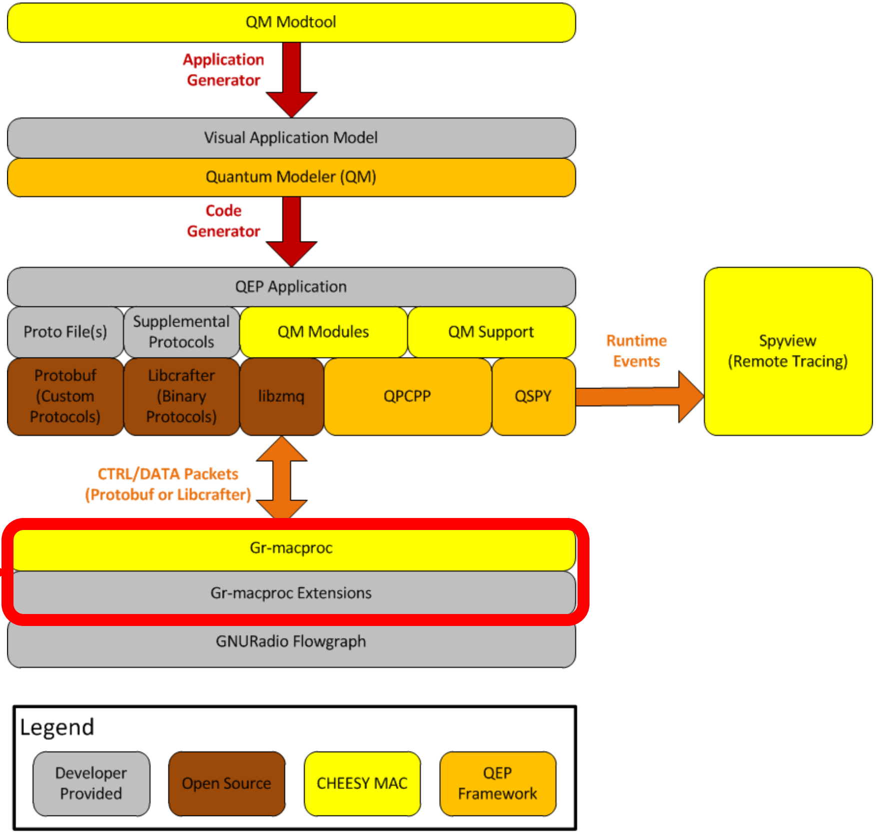
- CHEESY MAC is an Open Source toolkit which leverages the QP event driven design framework to enable a consistent approach to development of communication protocols
- CHEESY MAC is intended to interface with physical layer implementations to enable ease of development and maintenance in complex applications

The CHEESY MAC Toolkit

CHEESY MAC Architecture

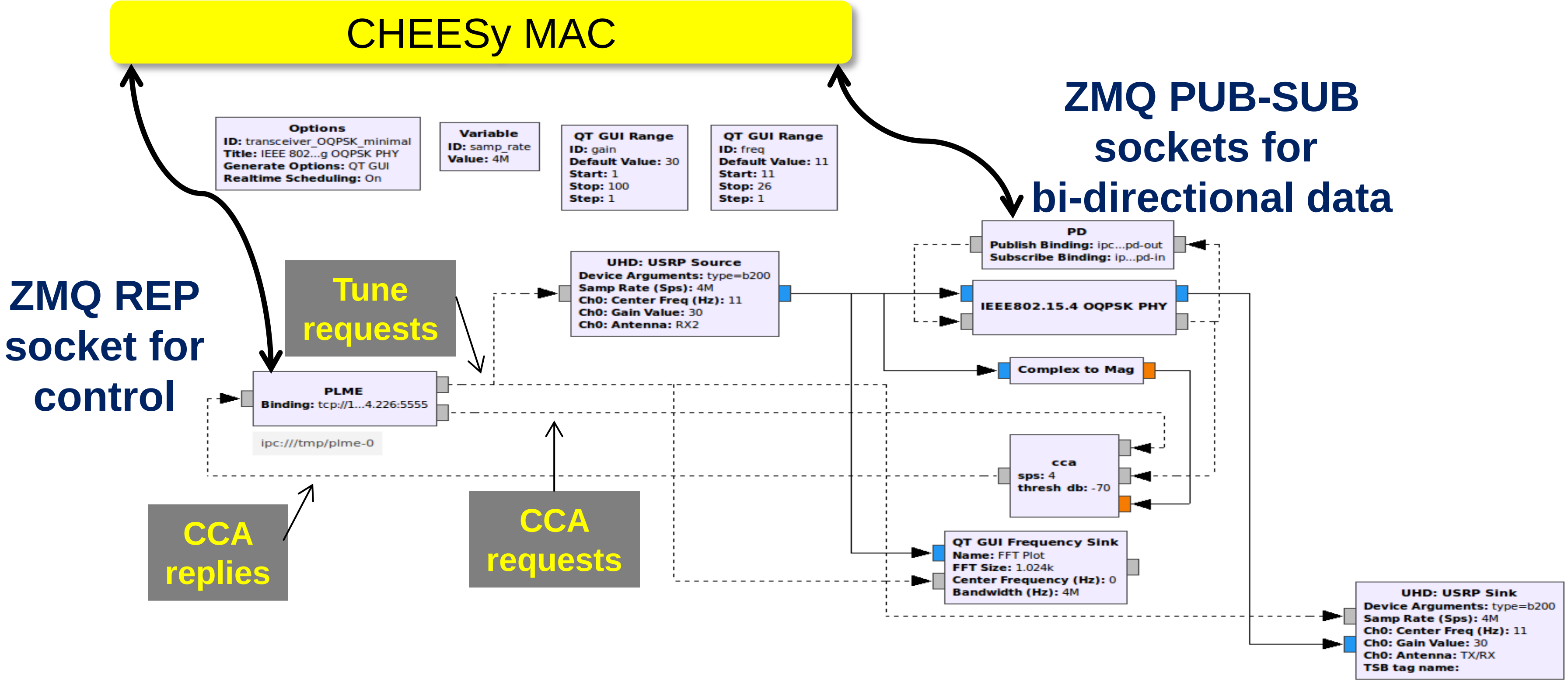


CHEESY MAC Architecture



Control

Control in *CHEESY MAC*



CHEESy MAC

ZMQ REP socket for control

ZMQ PUB-SUB sockets for bi-directional data

Tune requests

CCA replies

CCA requests

Options
ID: transceiver_OQPSK_minimal
Title: IEEE 802...g OQPSK PHY
Generate Options: QT GUI
Realtime Scheduling: On

Variable
ID: samp_rate
Value: 4M

QT GUI Range
ID: gain
Default Value: 30
Start: 1
Stop: 100
Step: 1

QT GUI Range
ID: freq
Default Value: 11
Start: 11
Stop: 26
Step: 1

PD
Publish Binding: ipc...pd-out
Subscribe Binding: ip...pd-in

UHD: USRP Source
Device Arguments: type=b200
Samp Rate (Sps): 4M
Ch0: Center Freq (Hz): 11
Ch0: Gain Value: 30
Ch0: Antenna: RX2

IEEE802.15.4 OQPSK PHY

Complex to Mag

PLME
Binding: tcp://1...4.226:5555

ipc:///tmp/plme-0

cca
sps: 4
thresh db: -70

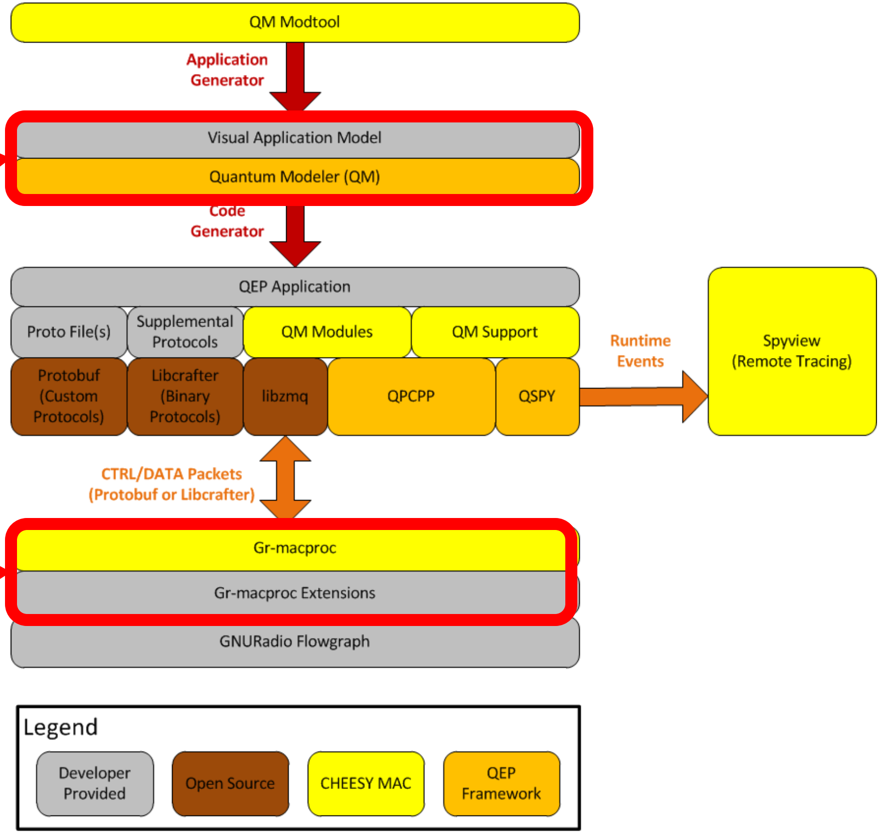
QT GUI Frequency Sink
Name: FFT Plot
FFT Size: 1.024k
Center Frequency (Hz): 0
Bandwidth (Hz): 4M

UHD: USRP Sink
Device Arguments: type=b200
Samp Rate (Sps): 4M
Ch0: Center Freq (Hz): 11
Ch0: Gain Value: 30
Ch0: Antenna: TX/RX
TSB tag name:

CHEESY MAC Architecture

Hierarchical State Machines

Control



QM Modtool

Application Generator

Visual Application Model

Quantum Modeler (QM)

Code Generator

QEP Application

Proto File(s)

Supplemental Protocols

QM Modules

QM Support

Protobuf (Custom Protocols)

Libcrafter (Binary Protocols)

libzmq

QPCPP

QSPY

Runtime Events

Spyview (Remote Tracing)

CTRL/DATA Packets (Protobuf or Libcrafter)

Gr-macproc

Gr-macproc Extensions

GNURadio Flowgraph

Legend

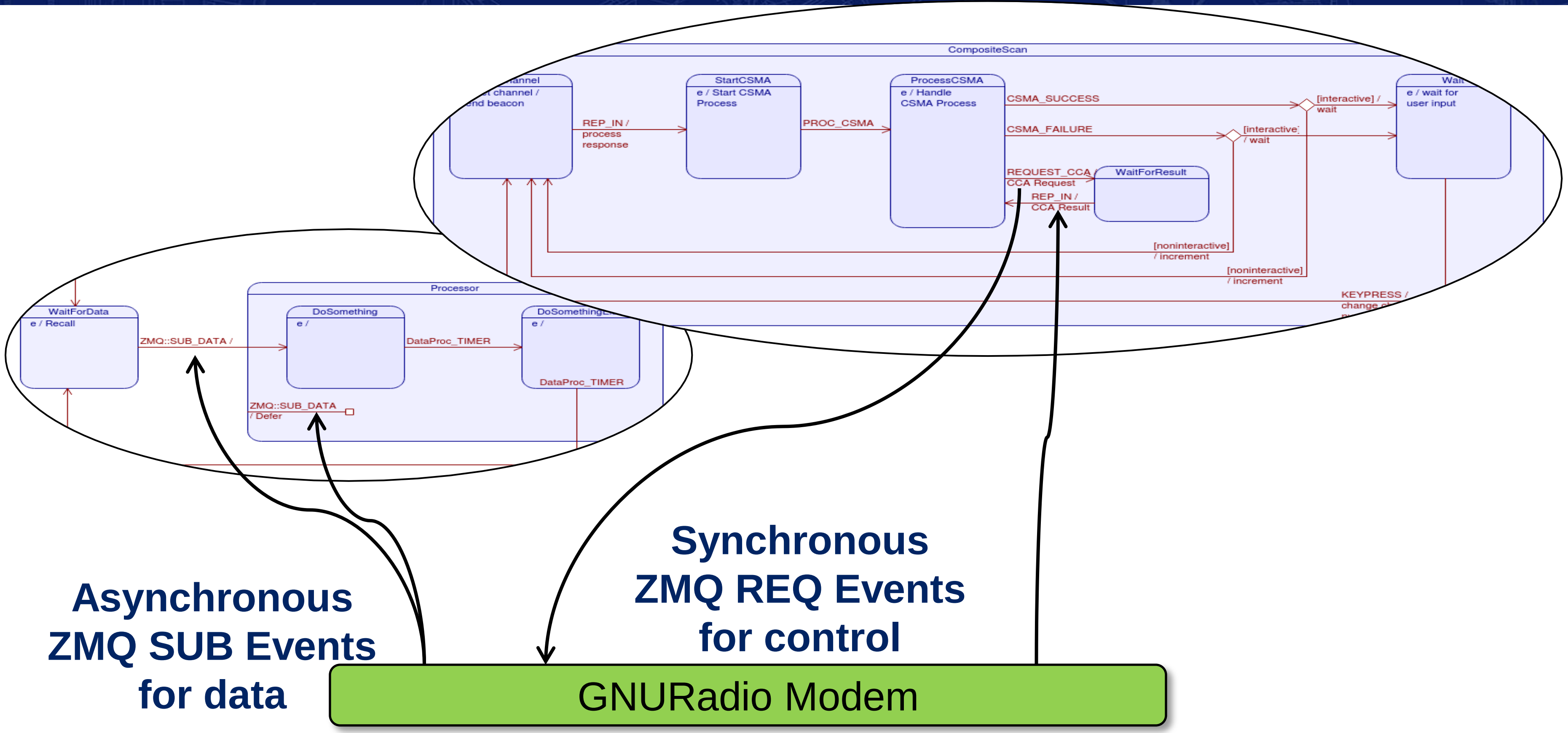
Developer Provided

Open Source

CHEESY MAC

QEP Framework

Hierarchical State Machines in CHEESY MAC



Asynchronous ZMQ SUB Events for data

Synchronous ZMQ REQ Events for control

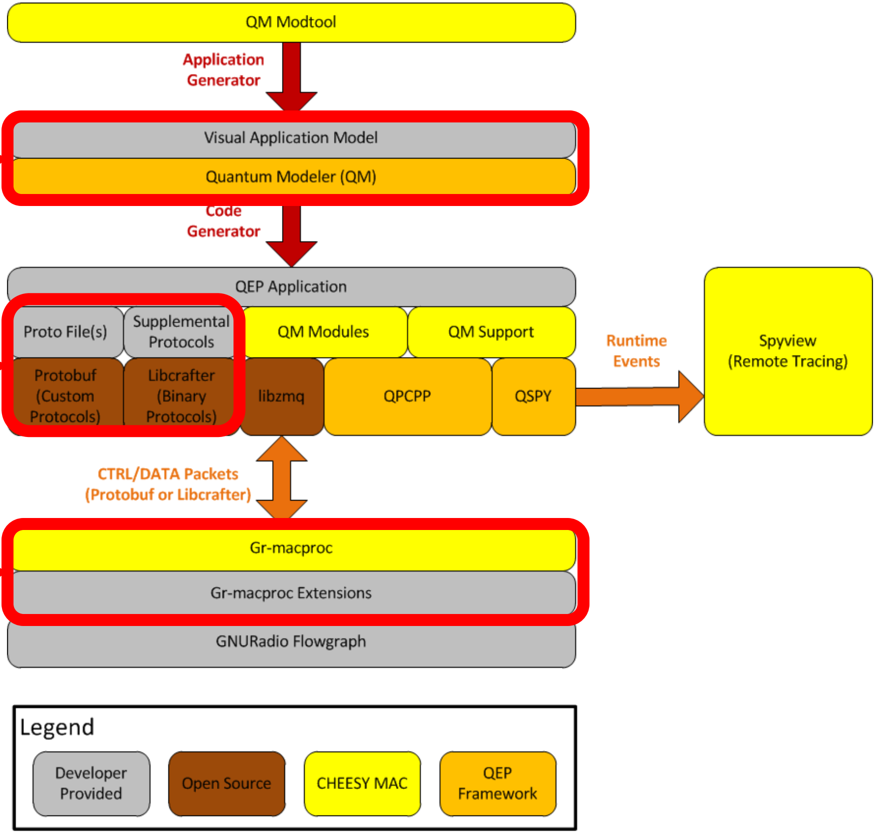
GNURadio Modem

CHEESY MAC Architecture

Hierarchical State Machines

Encapsulation

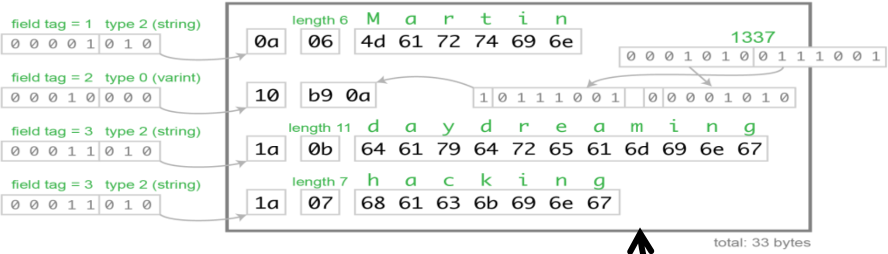
Control



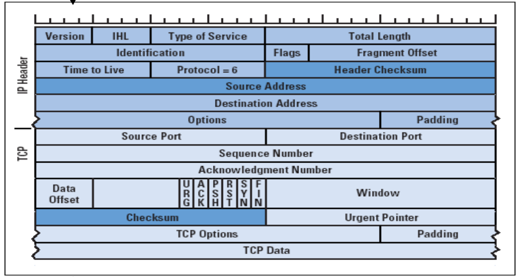
Encapsulation in CHEESY MAC

CHEESy MAC

Protobuf Encapsulation


field tag = 1 type 2 (string)
0 0 0 0 1 0 1 0
field tag = 2 type 0 (varint)
0 0 0 1 0 0 0 0
field tag = 3 type 2 (string)
0 0 0 1 1 0 1 0
field tag = 3 type 2 (string)
0 0 0 1 1 0 1 0
length 6 M a r t i n
0a 06 4d 61 72 74 69 6e
1337
0 0 0 1 0 1 0 0 1 1 1 0 0 1
10 b9 0a
1 0 1 1 1 0 0 1 0 0 0 0 1 0 1 0
length 11 d a y d r e a m i n g
1a 0b 64 61 79 64 72 65 61 6d 69 6e 67
length 7 h a c k i n g
1a 07 68 61 63 6b 69 6e 67
total: 33 bytes

Binary Encapsulation Using Libcrafter



GNURadio Modem

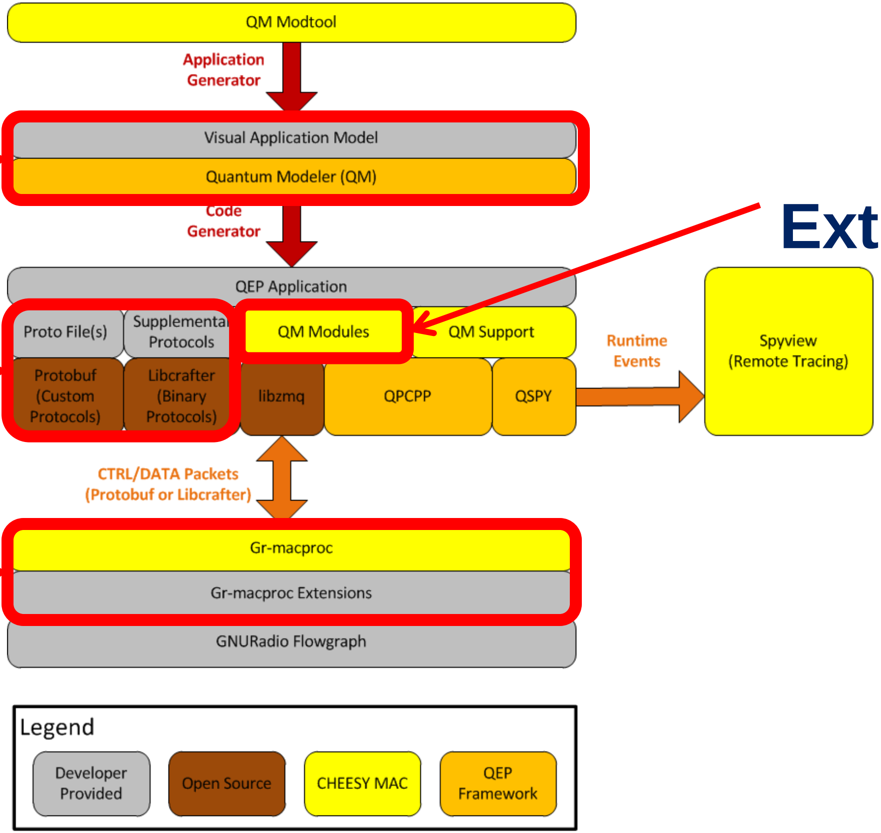
CHEESY MAC Architecture

Hierarchical State Machines

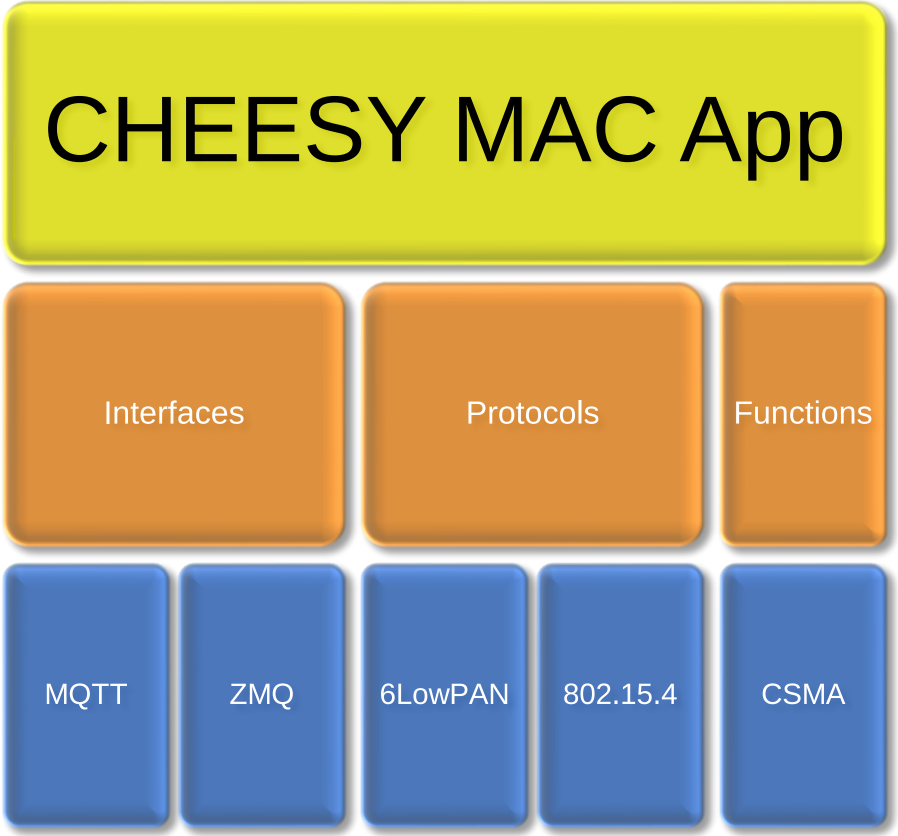
Encapsulation

Control

Extensions



Extensions in *CHEESY MAC*



CHEESY MAC App

Interfaces

Protocols

Functions

MQTT

ZMQ

6LowPAN

802.15.4

CSMA

- **Import existing QM packages**
  - **ZMQ uses an import package**
- **Create custom packages for extended functionality**
  - **Implement custom state machines in qm_modules**
  - **Import package into QM project**
  - **CHEES will automatically build and link new functionality**

CHEESY MAC Architecture

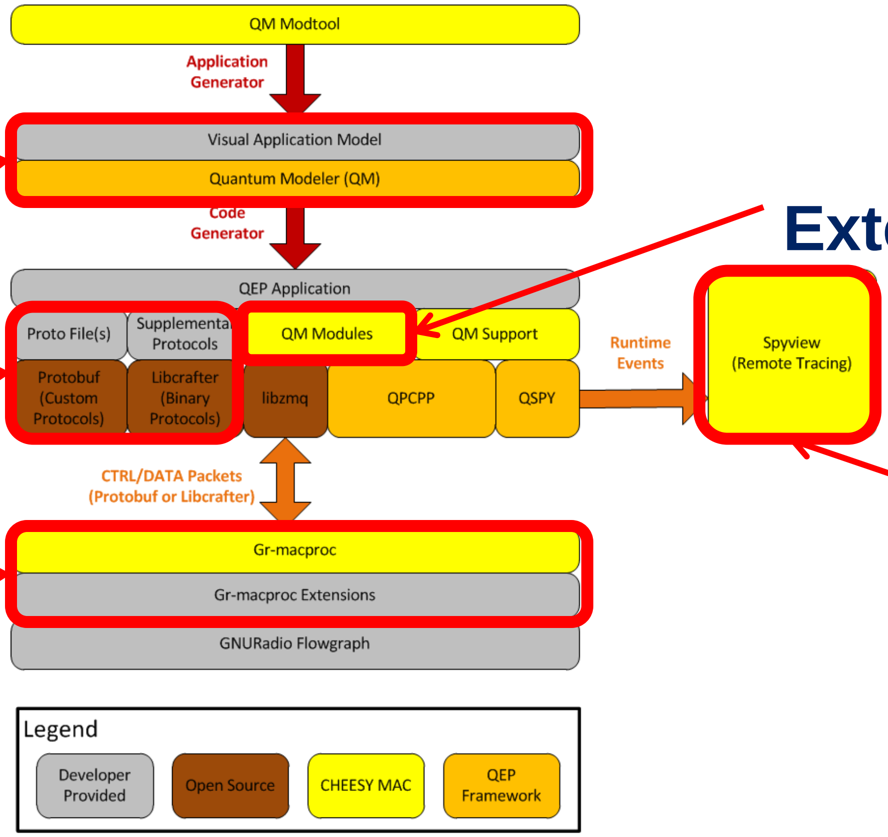
Hierarchical State Machines

Encapsulation

Control

Extensions

SPY



QM Modtool

Application Generator

Visual Application Model

Quantum Modeler (QM)

Code Generator

QEP Application

Proto File(s)

Supplementa Protocols

QM Modules

QM Support

Protobuf (Custom Protocols)

Libcrafter (Binary Protocols)

libzmq

QPCPP

QSPY

Runtime Events

Spyview (Remote Tracing)

CTRL/DATA Packets (Protobuf or Libcrafter)

Gr-macproc

Gr-macproc Extensions

GNURadio Flowgraph

Legend

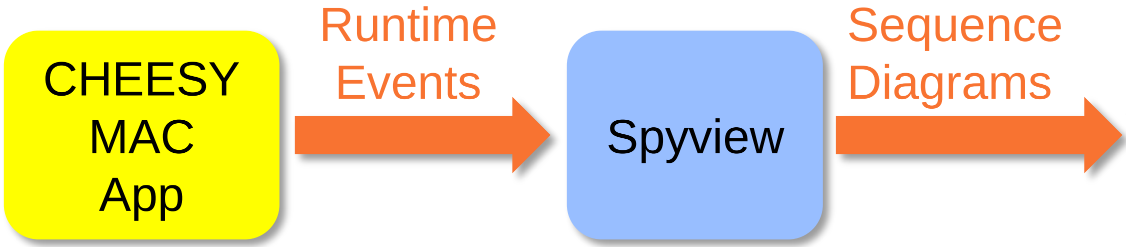
Developer Provided

Open Source

CHEESY MAC

QEP Framework

Spy in *CHEESY MAC*

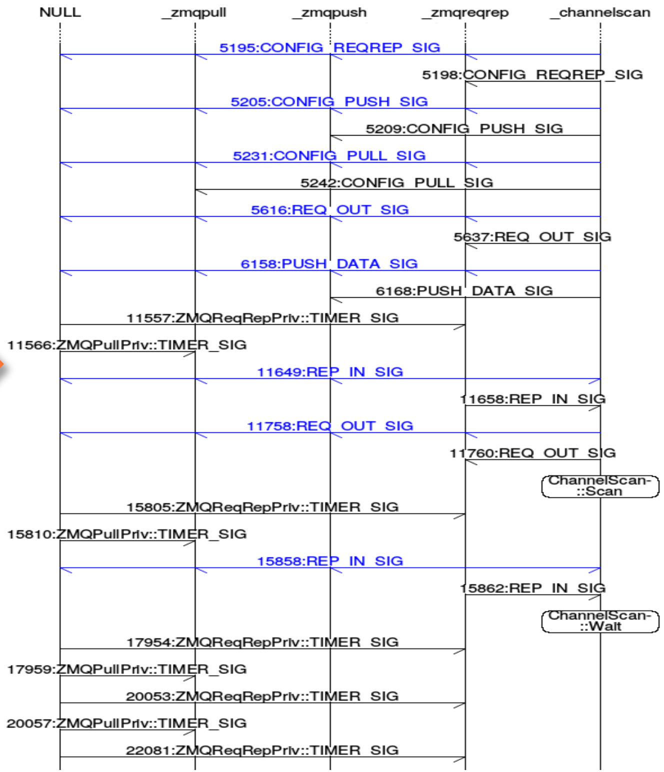


CHEESY MAC App

Runtime Events

Spyview

Sequence Diagrams



Source Code

- **Public Release Review Underway**

- **Will post to: https://github.com/cheesymactoolkit**

- **GPLv3 license**

APL

JOHNS HOPKINS

APPLIED PHYSICS LABORATORY