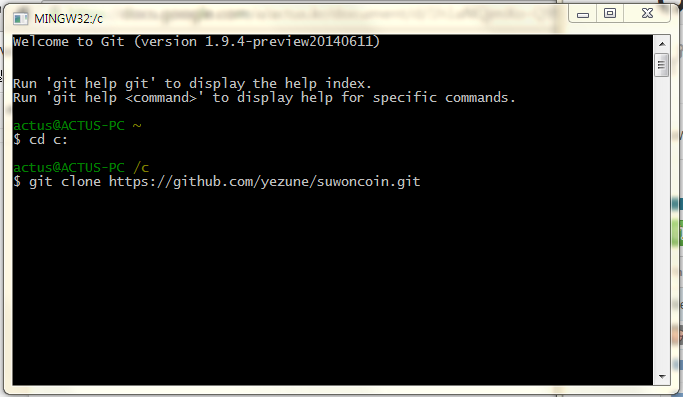
**suwoncoin window build**

**1.suwoncoin source download**

Git bash를 사용해서 github에 보관 중인 suwoncoin source를 다운 받습니다.



소스 관리 편의성을 위해 C 드라이브로 복사합니다. (물론 다른 경로에 다운로드 해도 상관 없습니다.)

**2. Install msys** <http://sourceforge.net/projects/mingw/files/Installer/mingw-get-setup.exe/download>에서 Mingw를 다운로드 받아 설치 합니다.

2.1.MinGW Manager 설치가 되면 All packages → MSYS mark 를 순서대로 클릭 합니다.

2.2 “**msys-base-bin msys-autoconf-bin msys-automake-bin msys-libtool-bin”** 을 선택해서 설치 한 다음 “**Apply changes”** 를 선택 합니다.

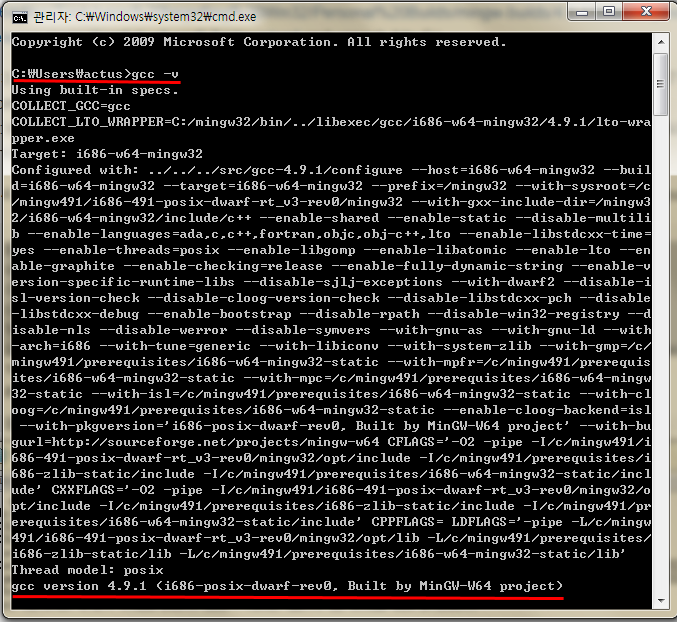
2.3 MinGW-build project toolchain 을 설치하기 위해 <http://sourceforge.net/projects/mingw-w64/files/Toolchains%20targetting%20Win32/Personal%20Builds/mingw-builds/4.9.1/threads-posix/dwarf/i686-4.9.1-release-posix-dwarf-rt_v3-rev0.7z/download> 를 클릭해서 파일을 다운 받은 다음 C 드라이브 아래에 압축 해제 합니다.

2.4 제어판 → 시스템 → 고급 시스템 설정 → 환경변수 순서로 클릭해서 mingw-builds와 관련된 bin 폴더들의 Path를 선언해 줍니다.

ex)C:\mingw32\bin;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\

※ Path에 들어간 디렉토리 이름과 디렉토리 이름이 다를 경우 동일하게 맞춰줍니다.

2.5 추가적으로 gcc 버전을 확인 합니다. (없으면 설치합니다.)



※1.5번까지 정상적으로 완료가 될 경우 C 드라이브 아래에 deps 폴더가 생성되어있고 이 폴더에 build 산출물들이 저장됩니다.

**3.1 OpenSSL**

<http://www.openssl.org/source/openssl-1.0.1i.tar.gz> 를 클릭해서 openssl-1.0.1i.tar.gz를 다운로드 합니다. C:\MinGW\msys\1.0 아래에 있는 msys.bat를 실행해서 다음 커맨드 라인을 입력 합니다.

cd /c/deps/  
tar xvfz openssl-1.0.1i.tar.gz  
cd openssl-1.0.1i  
Configure no-shared no-dso mingw  
make

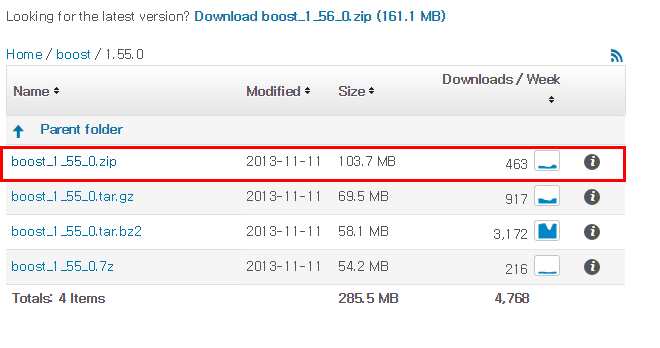
**3.2 Berkeley DB**

<http://download.oracle.com/berkeley-db/db-4.8.30.NC.tar.gz>에서 “db-4.8.30.tar.gz”를 다운 받아 압축 해제 한 후 configure and make Code를 입력 합니다.

cd /c/deps/  
tar xvfz db-4.8.30.NC.tar.gz  
cd db-4.8.30.NC/build\_unix  
../dist/configure --enable-mingw --enable-cxx --disable-shared --disable-replication  
make

**3.3 Boost**

<http://sourceforge.net/projects/boost/files/boost/1.55.0/> 에서 “boost\_1\_55\_0.zip” 파일을 클릭해서 다운 받습니다.



다운이 완료 되면 C 드라이브 아래 deps 디렉토리에 “boost\_1\_55\_0.zip” 압축해제 한 후

window command prompt를 실행해서 다음 커맨드 라인을 입력 합니다.

cd C:\deps\boost\_1\_55\_0\  
bootstrap.bat mingw  
b2 --build-type**=**complete --with-chrono --with-filesystem --with-program\_options --with-system --with-thread toolset**=**gcc variant**=**release link**=**static threading**=**multi runtime-link**=**static stage

컴파일을 하게 되면 boost libraries 를 요청하게 될 것입니다. 그리고 stage folder에 라이브러리 파일이 들어가게 됩니다. (C:\deps\boost\_1\_55\_0\stage)

**3.4 Miniupnpc**

<http://miniupnp.free.fr/files/download.php?file=miniupnpc-1.9.tar.gz> 를 클릭하면 miniupnpc.1.9.tar.gz 파일이 자동으로 다운로드 됩니다. 마찬가지로 C 드라이브에 있는 deps 폴더 안에 압축을 해제 하고 압축 해제한 폴더 이름을 miniupnpc로 rename 합니다.

그 다음 Windows command prompt 를 실행하여 다음과 같은 커맨드 라인을 입력 합니다.

cd C:\deps\miniupnpc  
mingw32-make -f Makefile.mingw init upnpc-static

**3.5 protoc**

<http://protobuf.googlecode.com/files/protobuf-2.5.0.zip> 를 클릭해서 “libpng-1.6.12.tar.gz”를 다운 받아 C 드라이브에 있는 deps 폴더 안에 압축을 해제 합니다. 압축 해제 후 msys shell을 실행하여 다음 커맨드를 입력 합니다.

cd /c/deps/protobuf-2.5.0  
configure --disable-shared  
make

**3.6 qrencode**

<http://prdownloads.sourceforge.net/libpng/libpng-1.6.12.tar.gz?download> 를 클릭해서 “miniupnpc-1.9 tar.gz”를 다운 받은 후 C 드라이드에 잇는 deps 폴더 안에 압축을 해제 한 다음 다음 커맨드 라인을 입력 합니다.

cd /c/deps/libpng-1.6.12  
configure --disable-shared  
make  
cp .libs/libpng16.a .libs/libpng.a

그 다음 [http://fukuchi.org/works/qrencode/qrencode-3.4.4.tar.g](http://fukuchi.org/works/qrencode/qrencode-3.4.4.tar.gz)z 클릭해서 “qrencode-3.4.4.tar.gz”를 다운 받은 후 C 드라이드에 잇는 deps 폴더 안에 압축을 해제하고 다음 커맨드 라인을 입력 합니다.

cd /c/deps/qrencode-3.4.4  
  
LIBS**=**"../libpng-1.6.12/.libs/libpng.a ../../mingw32/i686-w64-mingw32/lib/libz.a" \  
png\_CFLAGS**=**"-I../libpng-1.6.12" \  
png\_LIBS**=**"-L../libpng-1.6.12/.libs" \  
configure --enable-static --disable-shared --without-tools  
  
make

**3.7 Qt 5 libraries**

qt는 반드시 ssl과 zlib 지원이 구성되어 잇어야 합니다. <http://download.qt-project.org/official_releases/qt/5.3/5.3.1/submodules/qtbase-opensource-src-5.3.1.7z> 와 <http://download.qt-project.org/official_releases/qt/5.3/5.3.1/submodules/qttools-opensource-src-5.3.1.7z> 를 클릭해서 Qt base와 tools sources를 다운 받아 C 드라이브 밑에 압축 해제 합니다. 압축 해제 후 windows command prompt 를 실행하여 다음 커맨드라인을 입력 합니다.

set INCLUDE**=**C:\deps\libpng-1.6.12;C:\deps\openssl-1.0.1i\include  
set LIB**=**C:\deps\libpng-1.6.12\.libs;C:\deps\openssl-1.0.1i  
  
cd C:\Qt\5.3.1  
configure.bat -release -opensource -confirm-license -static -make libs -no-sql-sqlite -no-opengl -system-zlib -qt-pcre -no-icu -no-gif -system-libpng -no-libjpeg -no-freetype -no-angle -no-vcproj -openssl -no-dbus -no-audio-backend -no-wmf-backend -no-qml-debug  
mingw32-make  
set PATH**=**%PATH%;C:\Qt\5.3.1\bin  
cd C:\Qt\qttools-opensource-src-5.3.1  
qmake qttools.pro  
mingw32-make

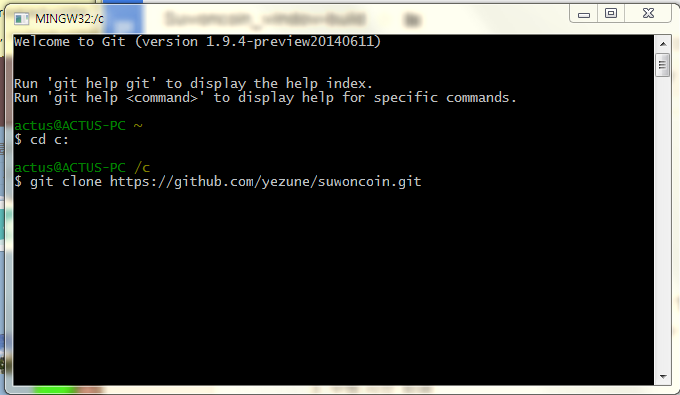
※ mingw32-make 에 -j 옵션을 주게 되면 속도가 더 빨라집니다. cpu가 쿼드코어 일 경우 -j4 or -j5 옵션을 주게 될 경우 최고의 결과를 얻을 수 있습니다.

**4. 수원 시민 화폐 적용**

4.1 git bash를 다운 받아 설치 해서 git clone으로 수원 시민 화폐 source를 다운 받습니다.

git bash 다운로드 경로 : <http://git-scm.com/downloads>

다운 & 설치가 완료 되면 git bash에서 git clone [https://github.com/yezune/suwoncoin.git](https://github.com/yezune/suwoncoin.git을) 을 입력해서 수원 시민 화폐 소스를 다운로드 받습니다.



※ 수원 시민 화폐 source 다운로드 예시

4.2 Release

window command prompt를 실행해서 다음 커맨드 라인을 입력 합니다.

set PATH=%PATH%;C:\Qt\5.3.1\bin

cd C:\suwoncoin\

qmake "USE\_QRCODE=1" "USE\_UPNP=1" "USE\_IPV6=1" suwoncoin-qt.pro

mingw32-make -f Makefile.Release

**5.suwoncoin\_window\_build 확인**

제일 첫번째 순서에서 git bash를 통해 다운 받은 suwoncoin 폴더에 가보면 “release” 폴더가 있습니다.

